LOAN DOCUMENT

PHOTOGRAPH THIS SHEET

DTIC ACCESSION NUMBER

LEVEL

INVENTORY

DOCUMENT IDENTIFICATION

DISTRIBUTION STATEMENT

ACCESSION FOR
NTIS GRA&I ☒
DTIC TRAC ☐
UNANNOUNCED ☐
JUSTIFICATION

BY
DISTRIBUTION/
AVAILABILITY CODES

DISTRIBUTION	AVAILABILITY AND/OR SPECIAL	
A-1		

DISTRIBUTION STAMP

HANDLE WITH CARE

DATE ACCESSIONED

Reproduced From
Best Available Copy

DATE RETURNED

19981223 079

DATE RECEIVED IN DTIC

REGISTERED OR CERTIFIED NUMBER

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-FDAC

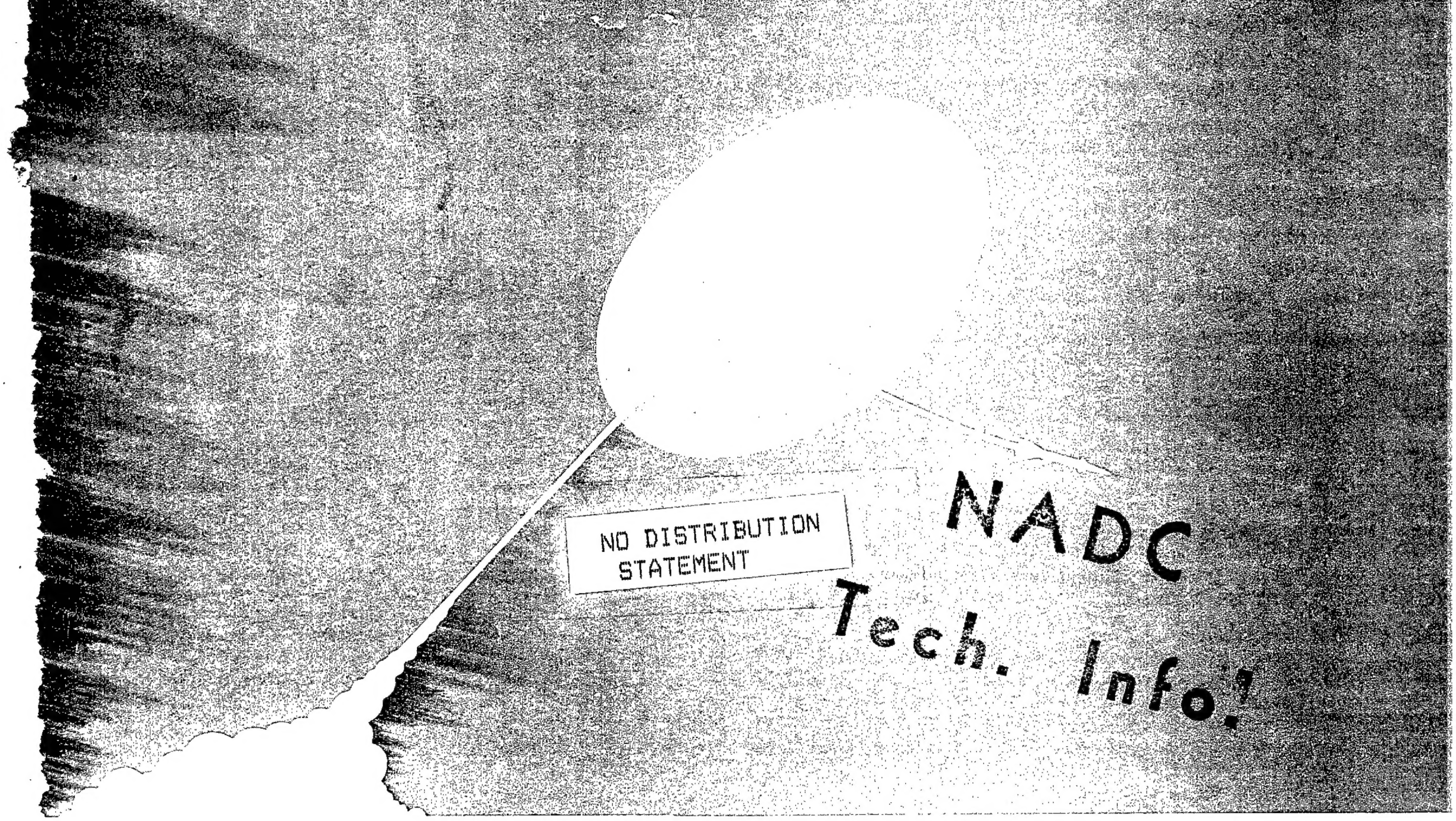
DTIC FORM JUN 90 70A

DOCUMENT PROCESSING SHEET

PREVIOUS EDITIONS MAY BE USED UNTIL STOCK IS EXHAUSTED.

LOAN DOCUMENT

UNCLASSIFIED



APPENDIX 26

SIGNAL SORTER (SS) SUPERVISOR DESIGN SPECIFICATION & FLOW DIAGRAMS

FINAL SOFTWARE REPORT

DATA ITEM NO. A005

INTEGRATED ELECTRONIC WARFARE SYSTEM ADVANCED DEVELOPMENT MODEL (ADM)

780987-26

PREPARED FOR:

NAVAL AIR DEVELOPMENT CENTER
WARMINSTER, PENNSYLVANIA

CONTRACT N62269-75-C-0070

RAYTHEON
ELECTROMAGNETIC
SYSTEMS DIVISION

1 OCTOBER 1977

UNCLASSIFIED

APPENDIX 26

SIGNAL SORTER SUPERVISOR DESIGN SPECIFICATION

FINAL SOFTWARE REPORT

DATA ITEM A005

INTEGRATED ELECTRONIC WARFARE SYSTEM (IEWS)
ADVANCED DEVELOPMENT MODEL (ADM)

Contract No. N62269-75-C-0070

Prepared for:

Naval Air Development Center
Warminister, Pennsylvania

Prepared by:

RAYTHEON COMPANY
Electromagnetic Systems Division
6380 Hollister Avenue
Goleta, California 93017

1 OCTOBER 1977

A30

RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173	CODE IDENT NO. 49956	SPEC NO. CG-983645 - 3 SHEET 1 OF	REV

TYPE OF SPEC

COMPUTER PROGRAM DESIGN SPECIFICATION

TITLE OF SPEC

IEWS Signal Sorter Supervisor Software

FUNCTION	APPROVED	DATE	FUNCTION	APPROVED	DATE
WRITER	N. Fujiyoshi	11/12/75			

REVISIONS

CHK	DESCRIPTION	REV	CHK	DESCRIPTION	REV

REVISION																					
SHEET NO.																					

REV STATUS OF SHEETS	REVISION																		
	SHEET NO.																		

1.0 SCOPE

This document describes the design specifications of the IEWS Signal Sorter Supervisor Software.

2.0 APPLICABLE DOCUMENTS

ESD-SB-001	Signal Sorter Integrated Electronics Warfare System, Rev. 4
WS-8506	Requirements for Digital Computer Program Documentation, Rev. 1, dated 1 November 1971.
CG-983645	IEWS Signal Sorter Computer Program Performance Specification
CG-983645	IEWS Signal Sorter NESU Software
5413-IEWS:75:03	IEWS Input Buffer Functional Specification, Rev. B
5413-IEWS:75:05	IEWS Signal Sorter Supervisor Design Specification, Rev. B
5413-IEWS:75:06	IEWS Signal Sorter Track Correlator Design Specification, Rev. A
RP-16	Microprocessor Manual

3.0 REQUIREMENTS

3.1 FUNCTION ALLOCATION/DESCRIPTION

The Supervisor consists of the following modules:

- Initialization
- Task Manager
- Core Manager
- Update
- NESU Message Handler
- SC Message Handler
- Aux. Functions

These modules make up a multi-task priority real-time operating system in which each of the functions of the Supervisor is performed by one or more tasks.

3.1.1 INITIALIZATION MODULE

The Initialization module performs the initialization of the Supervisor software and hardware including the Input Buffer, the FIFO, and the Track Correlator. This module is used both at initial program load time and upon receipt of a SC Initialization Command.

3.1.2 TASK MANAGER MODULE

The Task Manager Module performs the function of task scheduling and dispatching according to priority. The module consists of two subroutines: A Task Scheduler and a Task Dispatcher. These subroutines maintain four task queues, each queue corresponding to a different task priority. Tasks are placed on the queues by other modules in the system by calling the Task Scheduler and giving it a Task Control Block (TCB) and a priority number. Initiation of execution of a task is done by searching for a non-empty queue starting with the highest priority one. If more than one TCB is contained on a queue, they are handled in a First-In-First-Out basis. Control is then transferred to the address contained in the first TCB found.

3.1.3 CORE MANAGER MODULE

The Core Manager Module performs the maintenance of available core blocks for the Supervisor modules. This module consists of two subroutines: a get core block routine and a return core block routine. Each core block consists of five contiguous words and is initially placed in the available core block queue common to both the Supervisor and the NESU. These blocks are used for TCB's and for

storing PDW's. Blocks are obtained by the modules by calling the get core block routine, and are returned to the available core block queue by calling the return core block routine.

3.1.4 UPDATE MODULE

The Update Module performs the updating of all track files in the Track Correlator. This module consists of a Schedule Update subroutine and four tasks: Initiate Update, Start Update, Update Track and Time-out Check. The Update module maintains four update queues, each queue corresponding to a different update priority. Each entry in the queues consists of two words, one for each track file to be updated. Entries are placed on the queues by calling the Schedule Update routine. This routine in turn schedules the Start Update task if less than fice updates are in progress. The Start Update task searches the update queues for the highest priority update scheduled, initializes the Emitter Table Entry, and sets the count in the Track Correlator for the file to be updated. The Initiate Update task is scheduled periodically by the Real Time Clock Interrupt Handler. It schedules track files for update depending on category and last time of update. The Update Track is scheduled by the NESU interrupt handler whenever the specified number of PDW's have been received for a track file to be updated. This task performs the actual updating of the track files in the Track Correlator. The Time-Out Check task checks for track files in the update process which have not received any PDW's within the trap time.

3.1.5 NESU MESSAGE HANDLER MODULE

The NESU Message Handler module processes all messages generated and received from the NESU. This module consists of the NESU Interrupt Handler, the New Track Start task, and the low level NESU Message handler. The NESU Interrupt Handler processes the NESU generated interrupt signifying that a high level message has been

placed in one of the two high level message buffers. The high level message is either a New Emitter Start message or a Supervisor PDW message. If it is a New Emitter Start message, the NESU Interrupt Handler schedules the New Track Start task. If the message is a Supervisor PDW, the interrupt handler links it to the proper track file.

The low-level NESU message handler is a task scheduled by the Message Polling task and processes all of the low level messages generated and sent by the NESU. These consist of the CAM file dump and the AOA file dump.

3.1.6 SC MESSAGE HANDLER MODULE

The SC Message Handler Module processes all of the messages generated and sent by the SC. This module consists of the SC Interrupt Handler and the SC Message Handler. The SC Interrupt Handler processes the SC interrupt which signifies a high level message from the SC. These messages consist of the Start, Pause, Initialize, Pause NESU, and Initialize and Start NESU Commands. Depending on the message, the SC interrupt handler sets the necessary flags and/or schedules or transfers control to the necessary routines. The SC Message Handler processes all low level SC messages and is scheduled by the Message Polling task.

3.1.7 AUXILLIARY FUNCTIONS

The auxilliary functions consist of the Message Polling task and the test of the interrupt handlers. The Message Polling task checks both the SC and NESU low level message buffers for an incoming message. If a message is present, it schedules the SC Message Handler or the NESU Message Handler which perform the message processing. The Real Time Clock Interrupt Handler maintains the system clock which contains current time. Each "tick" corresponds to 250 Oms.

On each interrupt the RTC Interrupt Handler sets the NESU purge flag and schedules the Time-Out Check task. Every fourth interrupt, or once per second it schedules the Initiate Interrupt task. The Bus Hung, Watchdog Timer, and Panic Button interrupts cause the Supervisor to send an error message to the SC and halt. The IB less than 1/4 full and greater than 3/4 full cause a TBD.

3.1.8 TASK STRUCTURE

Figure 3.1 shows the assignment of tasks by priority level with level 0 being the highest priority. The task priority structure is a software extension of the hardware interrupt structure allowing a more modular and self-contained design of independent modules. The Task Manager acts as a system utility allowing tasks to schedule other tasks without regard as to whether any higher priority tasks are waiting execution. A given task may also be scheduled many times before it is executed the first time.

The Message Polling task runs as a background task continually scheduling itself. When a message is detected in one of the input message buffers, it schedules either the EC or NESU Message Handler. The execution of the other tasks is initiated by an interrupt from either the RTC or NESU.

3.2 FUNCTIONAL DESCRIPTION

3.2.1 INITIALIZATION MODULE

The Initialization module is first started by the SC which per forms an Initialize and New Start sequence with the address of the Initialization module (INITA) in location zero. The Initialization module in turn performs an Initialize and New Start sequence on the NESU microcontroller with the address of its Initialization module in its location zero. It then initializes and clear the files in the Input Buffer and the Track Correlator and all of the files and queues

in the Supervisor. It then sets the Idle flag, schedules the Message Polling task and enables interrupts. The same sequence of initialization is followed when the SC Interrupt Handler detects a Initialization command from the SC.

3.2.2 TASK MANAGER

A task is scheduled by executing the following calling sequence:

JSUB (=SCHED)

The X register must contain the address of the TCB and the B-register must contain the priority level. The TCB consists of five words:

Word 0	Used by task manager for linking
Word 1	Task entry point
Word 2-4	Used by scheduling routine to pass parameters to task

Each task is responsible for returning the TCB to available storage by calling the Return Block routine. When a task has completed execution it must execute a jump to the Task Manager as follows:

JUMP (=DISP)

The Dispatcher then transfers control to the highest priority task awaiting execution using the second word in the TCB and passes the TCB address to the task in the X-register.

The Task Manager maintains four pairs of pointers for the four task queue, one queue per priority level. Each pair consists of a Start of Queue (SOQ) pointer and an End of Queue (EOQ) pointer. The SOQ pointer contains the address of the first TCB while the EOQ pointer contains the address of the last TCB in the queue. If the queue is empty, both pointers are set to zero. The first word in each TCB contains the address of the next TCB in the queue except for the last TCB whose first word contains zero. The Scheduler uses the task priority level in the B-register to select the proper EOQ pointer which contains the address of the last TCB. The new TCB address is

is placed in the first word of that TCB and in the EOQ pointer, and the first word of the new TCB is set to zero. The Dispatcher searches for the first SOQ pointer which is non-zero starting with the pointer corresponding to priority level 0. The address in the SOQ pointer is saved in the X-register and the contents of the first word of the TCB is placed in the SOQ pointer. The Dispatcher then jumps to the address contained in the second word of the TCB.

3.2.3 CORE MANAGER

A block of storage is obtained by executing the following call:

```
JSUB          (=GTBL)
```

The Core Manager returns to the calling routine with the address of the Five word core block in the X-register. One or more core blocks are returned to free storage by executing the following call:

```
JSUB          (=RTBL)
```

The X-register must contain the address of the first block and the B-register must contain the address of the last block. The first word in each block must contain the address of the next block except for the last block which must have a zero in its first word. If only one block is being returned, both registers must contain the address of the block and the first word of the block must contain zero.

The Core Manager maintains a queue of available core blocks which is shared with the NESU Core Manager. The SOQ and EOQ pointers and the core blocks reside in a common 4K RAM and are initialized by the Supervisor Initializer. On each call to Get Block, a block is removed from the queue and passed to the calling routine. On a Return Block call, the returned blocks are linked to the queue.

3.2.4 UPDATE MODULE

The update of a track file is begun by calling the Schedule Update subroutine as follows:

JSUB (=SCUP)

with the track number in the A-register. This subroutine is called by the Start New Track task and the Initiate Update task.

The Schedule Update subroutine uses the track number to select the proper Emitter Table Entry. The emitter is then placed on the update queue depending on the New Emitter flag bit and the Update Priority bits. The Update Queue consists of four FIFO queues, each queue corresponding to a different priority; New Emitter, high data rate - threat, high data rate, and low data rate. There is a SOQ and a EOQ pointer for each queue which point to the first and last entries in the queue, respectively. Each entry in the queue consists of the last two words in the Emitter Table entries. If less than five updates are in progress, the Schedule Update subroutine schedules the Start Update task.

The Initiate Update task is scheduled by the RTC Interrupt Handler once per second (every four clock ticks). This task searches the Emitter Table for all emitters which are due for update. For each emitter found, the Initiate Update task calls the Schedule Update Subroutine which places the emitter on the Update Queue. An emitter is considered due for update if the flag bits indicate that the emitter entry is in use, not in update, the don't update bit is set, and the current time minus the last PDW/update time is equal or greater to the priority.

The Start Update task is scheduled by the Schedule Update subroutine and the Update task. This task searches the Update Queue for the highest priority entry and sets TCOUNT for that emitter in the Track Correlator Track Data Memory. If the New Emitter Flag bit is set in the Emitter Table Entry, the count is set to 31_{10}, otherwise it is set to 11_{10}. The PDW count in the Emitter Table entry is also set to 30_{10} or 10_{10} and the in update flag bit it set. The entry is removed from the Update Queue and the Update Count is incremented. If the Update Count is less than five, the Update Queue

is searched for the next highest priority entry and the process repeated.

The Update task is scheduled by the NESU Interrupt Handler when the last PDW has been received for updating an emitter. The third word of the TCB used to schedule the Update task must contain the track number of the emitter to be updated. If the New Emitter flag bit in the Emitter Table entry is set, all parameters are updated, otherwise only Pulse Width, Frequency, and PRI are updated. The new parameters are written into the TDM, and the In Update, New Emitter, and Time-Out flag bits are cleared. If the Throttled flag bit is set, the Throttle File entry in the Input Buffer is also updated. If the update priority of the Emitter Table entry is one, a Pulse Train Descriptor Word Message is sent to the SC.

The Time-Out Check task is scheduled by the RTC Interrupt Handler every 500 ms. This task searches the Emitter Table for all entries in the update process which have not received a PDW within the trap time. The trap time is 500 ms. for high data rate emitters and the SC specified purge time for all others. All high data rate emitters found will have their priority set to a low data rate emitter. Any low data rate emitter found will have its time-out flag set. The next time the Time-Out Check task is scheduled, it will send a File Delete Request Message to the SC for each entry with its time-out flag set, will return any core blocks used to store PDW's, and will set the don't update flag and clear the time-out flag.

3.2.5 NESU MESSAGE HANDLER MODULE

The NESU Interrupt Handler processes all NESU Interrupts which signal the presence of a high priority message from the NESU in one of the two NESU Input Buffers. The message may be either a New Emitter Start Message or a Supervisor PDW message. If a New Emitter Start Message is received, the NESU Interrupt Handler schedules the Start New Track task. If the message is a Supervisor PDW message, the PDW

is linked to the Emitter Table entry specified by the track number. If the number of PDW's linked equals the PDW Count in the Emitter Table entry, the Update task is scheduled with the track number in the third word of the TCB.

There are two NESU high priority input buffers of ten words apiece. The first word is used as a flag whose contents have the following meaning:

0	Buffer Empty
1	New Emitter Start Message
2	Supervisor PDW Message

The format for a New Emitter Start Message is:

word 0	-----			
word 1	pointer to PDW list			
Word 2	TAZ			
Word 3	0			
word 4	TPRIA			
word 5	TPRIB			
word 6	TPW			
word 7	TQPRI	TQPW	TQF	TQAZ
word 8	TF			
word 9	TFAG	TCW		

The contents of words 2 through 9 are in the format of the TDM file. The format for a Supervisor PDW Message is:

word 0	2
word 1	Pointer to PDW block
word 2	File number
word 3-9	not used

The PDW block pointed to by word 1 has the following format:

word 0	not used
word 1-4	Normal PDW format

The flag word in the message buffer is set by the NESU when it places a message in the buffer, and it is reset to zero by the Supervisor when the message has been processed.

The Start New Track task gets an unused track number from the Track Correlator. This is used to write a new track file into the TDM using the information in the New Emitter Start Message and to initialize an Emitter Table entry. If no unused file exists, a TBD message is sent to the SC. If the emitter requires throttling, the throttle count is calculated and a throttle file number is requested from the Input Buffer. The throttle file is then written into the Input Buffer, the Throttle Table entry initialized and a Throttle Alert Message sent to the SC. If no vacant throttle file exists, a TBD message is sent to the SC. A New Emitter Alert message is sent to the SC, and the Schedule Update subroutine is called.

The NESU Message Handler is scheduled by the Message Polling task when a low priority message is found in NESU low priority message buffer.

The low priority messages consist of a CAM File Dump message and an AOA File Dump message. If a CAM File Dump message is found, the NESU Message Handler sends a CAM File Dump message to the SC. If an AOA File Dump message is found, an AOA Readout message is sent to the SC.

The NESU low priority message buffer consists of four words where the first word is a flag word whose contents indicate the following:

0	Buffer empty
1	CAM File Dump Message
2	AOA File Dump Message

The other three words contain the message indicated by the flag word. The format for a CAM File Dump Message is:

word 0	1
word 1	NESU CAM File Number

word 2	MSB=valid bit, bits 0-9 = frequency
word 3	bits 8-12 = azimuth, bits 0-5 = count

The format for an AOA File Dump Message is:

word 0	2
word 1	Cell number
word 2	PDW Count

The flag word is set by the NESU when it places a message in the buffer. The flag word is cleared to zero by the Supervisor when the message has been processed.

3.2.6 SC MESSAGE HANDLER MODULE

The SC Interrupt Handler processes all SC Interrupts and the high priority messages they signal. These messages consist of the Start Command, Pause Command, Initialize Command, Pause NESU Command and the Initialize and Start NESU Command. Following is a summary of the actions taken on receipt of each message:

Start Command	Clear Idle flag Send Start Message to NESU Enable PE/STE channel in Input Buffer
Pause Command	Set Idle flag Disable PE/STE channel in Input Buffer Send Pause message to NESU
Initialize Command	Jump to Initializer
Pause NESU Command	Send Pause message to NESU
Init. and Start NESU Command	Send Initialize message to NESU Send Start message to NESU

The SC high priority message buffer consists of 16 words where the first word contains a flag and a word count, and the second word contains a command code. The message formats and command codes are specified in the IEWS Signal Sorter Computer Program Performance Specification - CG-983645.

The SC Message Handler task is scheduled by the Message Polling task whenever a message is detected in the SC low priority input message buffer. This task processes all low priority SC messages. Following is a summary of actions taken on receipt of each low priority message:

CAM File Dump Command	Send CAM File Dump Message to NESU
AOA Readout Request	Send AOA Readout Request Message to NESU
File Dump Request	Send File Dump Request Message to NESU
UPDW Command	Enable transfer to NESU words to Aux. Interface in Track Correlator
Synthetic PDW	Stores Synthetic PDW into Input Buffer
NESU Track Threshold	Send Modify Track Start Threshold Message to NESU
Quality Bit Modification	Write quality bits into TDM for file specified
Purge Time Modify	Change priority in Emitter Table for file specified, change purge time
PTDW Request	Read Track file from TDM and send Pulse Train Descriptor Word to SC
SPDW Request	Set THRSC, TTAMP, and TCODE in TDM for file specified
SPDW Stop	Clear THRSC in the TDM for the file specified
NEPDW Request	Send TBD message to SC and return the core blocks to storage
Delete Track File	Clear the valid bit in the TDM and the used bit in the Emitter Table for the file specified

Frequency Modification	Write the frequency into the TDM file specified
PRI Modification	Write the PRIA and PRIB into the TDM file specified
Pulse Width Modification	Write the PW into the TDM file specified
Throttle File Modification	Write the throttle information into the Input Buffer Throttle File specified
AOA Threshold Modification	Send an AOA Threshold Modification message to the NESU
Create Track File	Write track file information into TDM file specified, initialize Emitter Table entry Send Confirm File Creation to SC

The SC low priority message buffer consists of sixteen words where the first word contains a flag and word count and the second word contains a command word. The formats and contents of the low priority SC messages are described in IEWS Signal Sorter Computer Program Performance Specification - CG-983645.

3.2.7 AUXILIARY FUNCTIONS

The Message Polling task is initially scheduled by the Initialization module. This task checks the SC low priority message buffer for an incoming message and shcedules the SC Message Handler task if the flag is set. It then checks the NESU low priority message buffer for an incoming message and schedules the NESU Message Handler if the flag is set. It then resets the Watchdog Timer and schedules itself.

The Real Time Clock Interrupt Handler processes all RTC Interrupts. These occur once every 250ms. At each interrupt, the handler increments the system time, sets the NESU purge flag, and schedules the Time-Out Check task. Every fourth interrupt it schedules the Initiate Update task. If the Idle flag is set, the handler none of these actions are taken and the handler simply does an interrupt return.

The Bus Hung Interrupt Handler processes the bus hung interrupt. When an interrupt occurs, the handler sends a TBD message to the SC, saves all registers and halts the Supervisor.

The Watchdog Timer Interrupt Handler processes watchdog timer interrupts in the same manner as the Bus Hung Interrupt Handler.

The Panic Button Interrupt Handler processes panic button interrupts in the same manner as the Bus Hung Interrupt Handler.

The IB Interrupt Handler processes IB less than 1/4 full and greater than 3/4 full interrupts. Occurrence of these interrupts causes a TBD action.

3.2.8 EMITTER TABLE

The Emitter Table is used by the Supervisor to maintain and update track files. There are 128 entries in the table, one entry for each track file in the TDM. Each entry consists of nine words in the following format:

word 0	Flags
word 1	Update PDW SOQ pointer
word 2	Update PDW EOQ pointer
word 3	Update PDW count
word 4	Last PDW/Update Time
word 5	New Track PDW pointer
word 6	Throttle File number
word 7	Update Queue word 1
word 8	Update Queue word 2 - file number

Flag Bits (if set):

15	in use (valid)
14	in update process
13	throttled file
12	don't update
11	new emitter
10	time-out
7 - 0	update priority

Words 1 and 2 point to the PDW's used for performing an update. Word 3 contains a count which is used to keep track of the number of PDW's in the update list. Word 4 contains the time (internal) of the last PDW or update and is used to decide when to start the next update. Word 5 points to the first PDW in the chain of ten PDW's used by the NESU to start a new emitter. Word 6 contains the Input Buffer file number for a throttled emitter. Words 7 and 8 are used as an entry on the Update Queue.

3.2.9 THROTTLE TABLE

The throttle Table is used to maintain the Input Buffer Throttle File. It contains eight 2-word entries, one for each file in the IB. The format of each entry is:

word 0	TDM File number
word 1	Throttle Count

3.3. STORAGE AND PROCESSING ALLOCATION

Table 3.1 summarizes the memory storage and processing time for the Supervisor routines.

ROUTINE	MEMORY SIZE	PROCESSING TIME
Initialization	300 words	10 ms
Scheduler	30 words	90 micro-seconds
Dispatcher	40 words	100 micro-seconds
Get Block	20 words	60 micro-seconds

ROUTINE	MEMORY SIZE	PROCESSING TIME
Return Block	20 words	60 micro-seconds
Schedule Update Sub-routine	30 words	90 micro-seconds
Initiate Update task	40 words	3 ms
Start Update task	80 words	150 micro-seconds
Update Track task	500 words	3 ms
Time-out Check task	40 words	3 ms
NESU Interrupt Handler	50 words	100 micro-seconds
Start New Track task	200 words	600 micro-seconds
NESU Message Handler	100 words	150 micro-seconds
SC Interrupt Handler	200 words	200 micro-seconds
SC Message Handler task	1000 words	120 micro-seconds
Message Polling task	30 words	250 micro-seconds
RTC Interrupt Handler	50 words	300 micro-seconds
Bus Hung Interrupt Handler	20 words	60 micro-seconds
Watchdog Timer Interrupt Handler	20 words	60 micro-seconds
Panic Button Interrupt Handler	20 words	60 micro-seconds
IB Interrupt Handler	TBD	TBD
Emitter Table	1152 words	N/A
Throttle Table	16 words	N/A
Free Core Storage	2450 words	N/A
TOTAL	6378 words	

TABLE 3.1

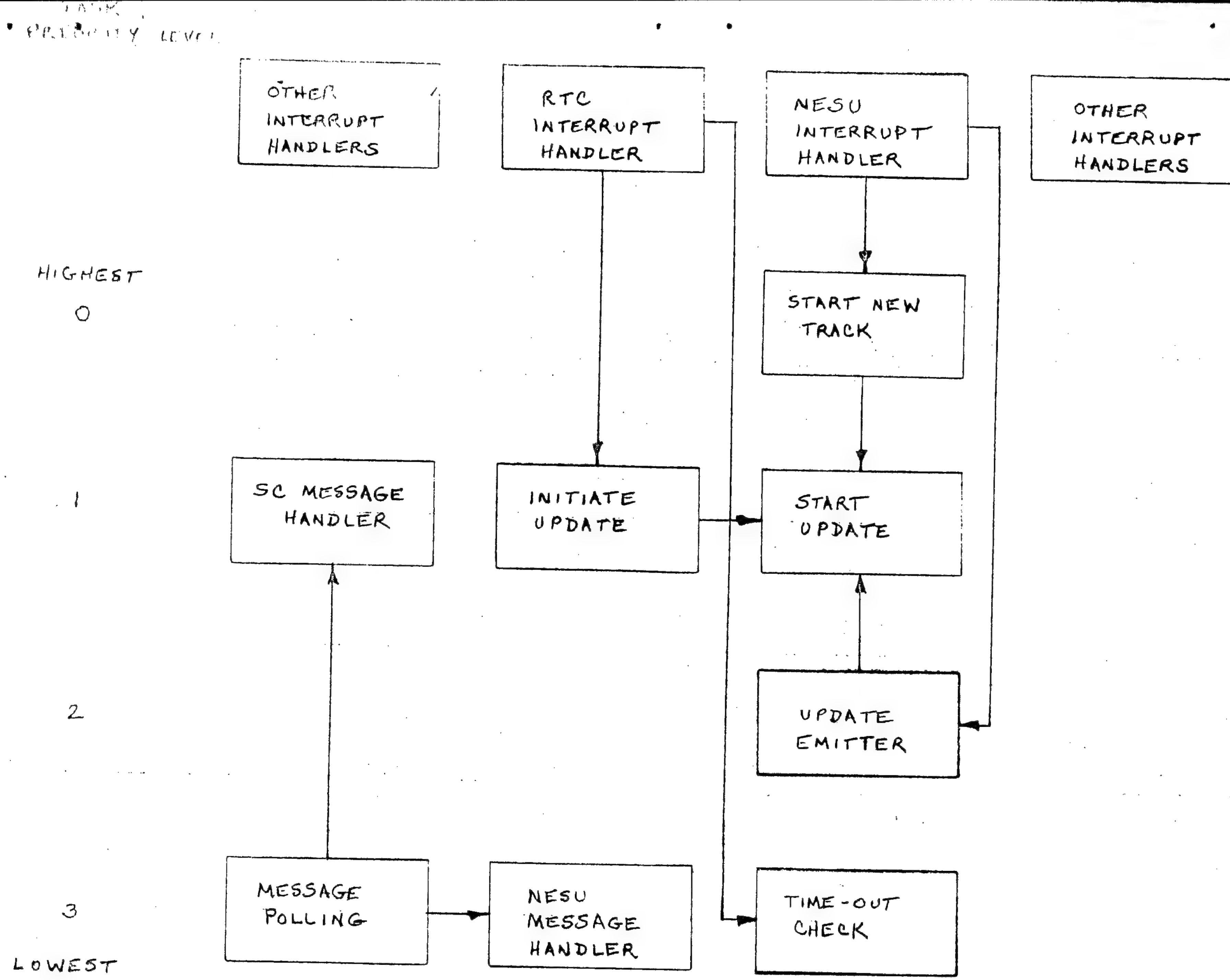
STORAGE AND PROCESSING TIME ALLOCATION

Processing speed of the Supervisor is a function of the number of updates being performed, the number of new emitters being started, and the number of messages being sent and received to and from the SC and the NESU. In a new-start situation where 30 emitters are being started and updated, the processing load per second is as follows:

NESU Interrupt Handler	93 ms
Start New Track task	18 ms
Schedule Update Subroutine	2.8 ms
Start Update task	4.5 ms
Update Emitter task	90 ms
Scheduler	8.1 ms
Dispatcher	9.0 ms
TOTAL	225.3 ms./sec.

In a steady-state situation where 60 emitters are being updated per second and 5 emitters are being started per second, the processing load is:

NESU Interrupt Handler	75.7 ms
Start New Track task	3.0 ms.
Schedule Update Subroutine	5.85 ms
Initiate Update task	3.0 ms
Start Update task	9.75 ms
Update Emitter task	195.0 ms
Scheduler	12.24 ms
Dispatcher	13.6 ms
TOTAL	318.13 ms./sec.



SUPERVISOR TASK SCHEDULING
FIGURE 3.1

3.4 FUNCTIONAL FLOW

Figure 3.1 shows the flow of control and task priority assignments in the Supervisor. Control is transferred from one task to another by scheduling as indicated by the arrows. All processing is initiated by either interrupts or receipt of a message. The interrupts received in order of priority are:

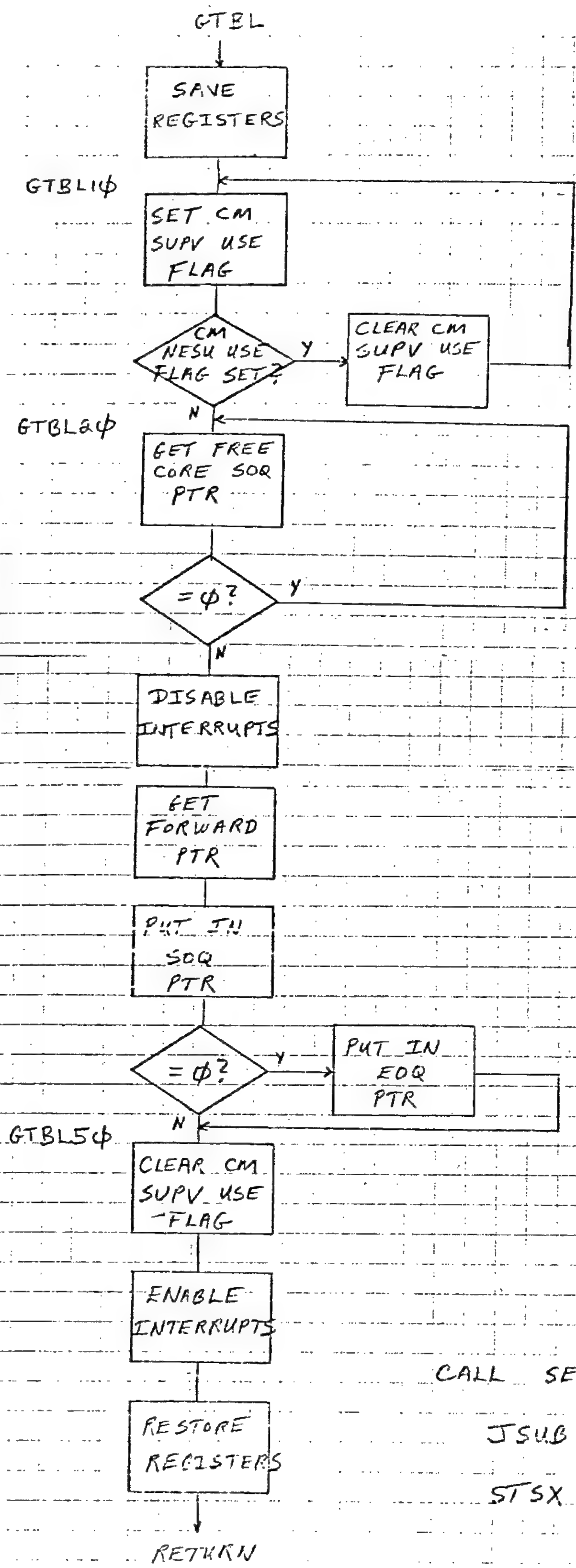
Bus Hung
Watchdog Timer
Panic Button
SC Message
New Emitter Alert
Real Time Clock
IB 3/4 full
IB 1/4 full

The Bus Hung, Watchdog Timer, and Panic Button interrupts normally indicate a hardware or software malfunction which is unrecoverable. The Bus Hung interrupt occurs if there is no response when the software attempts to address memory or a hardware unit. The Watchdog Timer interrupt occurs if the software does not reset the Watchdog Timer within a TBD interval. The Panic Button interrupt is caused by an operator either because the Sorter is malfunctioning or to stop the Sorter. The SC Message interrupt indicates the arrival of a high priority SC message and is caused by the SC. The New Emitter Alert indicates the presence of a high priority NESU message and is caused by the NESU. The Real Time Clock interrupt occurs every 250 ms. and is generated by an external Real Time Clock signal. The IB 3/4 full and IB 1/4 full interrupts are generated by the IB FIFO. Processing of the interrupts is detailed in paragraph 3.2.

3.5 PROGRAMMING GUIDELINES

The Supervisor software is written in RP-16 Assembly language and assembled using the RP-16 Relocatable Assembler (RAMA). The software is loaded into the Supervisor RAM by the SC which loads a loading routine into the common 1K RAM, Initializes the New Starts the Supervisor Micro-Processor, and transfers the object text records to the loading routine via the 1K RAM.

FLOW CHART



CALL SEQ –

JSUB (=GTBL)

STSX (ADDR OF BLOCK)

REMARKS

'CM' MEANS 'CORE MANAGER'

SUPPLY OF CORE BLOCKS EXHAUSTED

RAYTHEON

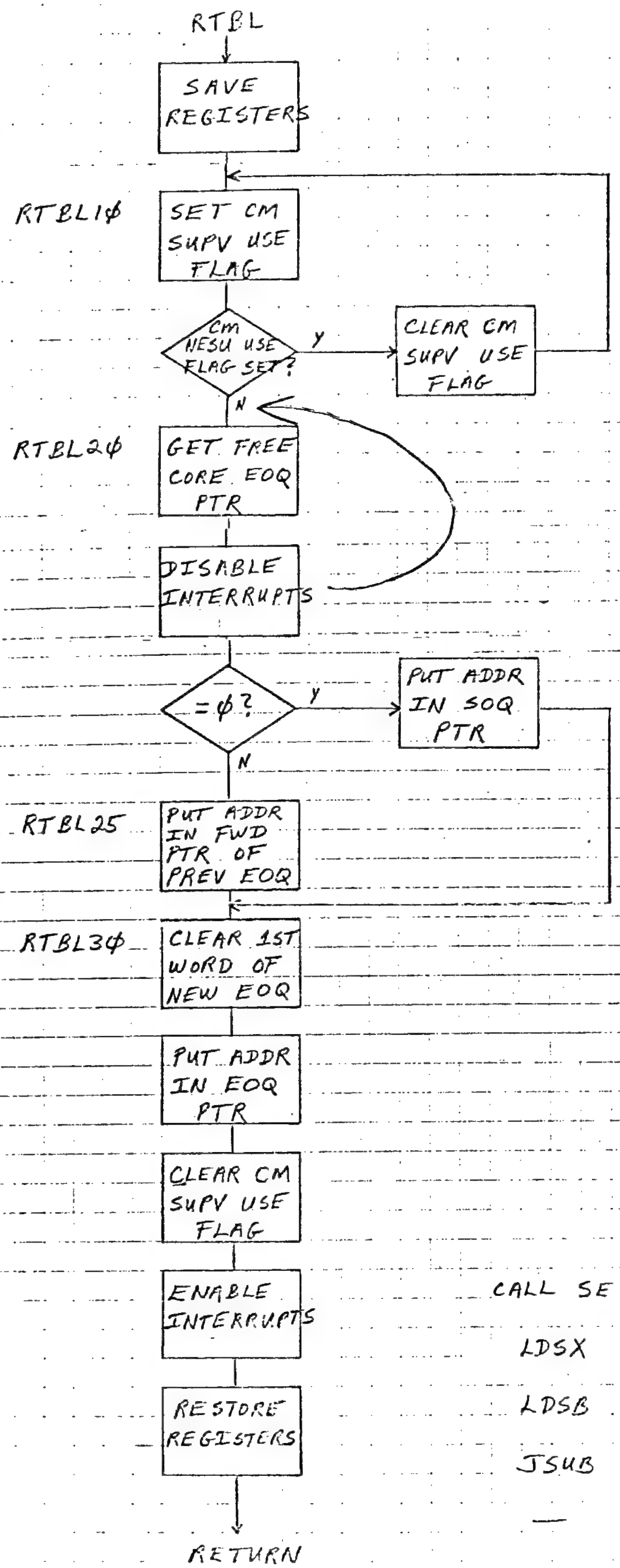
RAYTHEON COMPANY
LEXINGTON, MASS 02173

PROGRAM ROUTINE: [illegible] SUPV

GET BLOCK SUBROUTINE

49956 | T. [illegible] | 1 JUN 76

NUMBER 2 | SHEET 1 OF 1



'CM' MEANS 'CORE MANAGER'

CALL SEQ —

LDSX (ADDR OF 1ST BLK RETURNED)

LDSB (ADDR OF LAST BLK RETURNED)

JSUB (=RTBL)

— NORMAL RETURN

RAYTHEON

RAYTHEON COMPANY

LEXINGTON, MASS 02173

ILWS SORTER SUPV

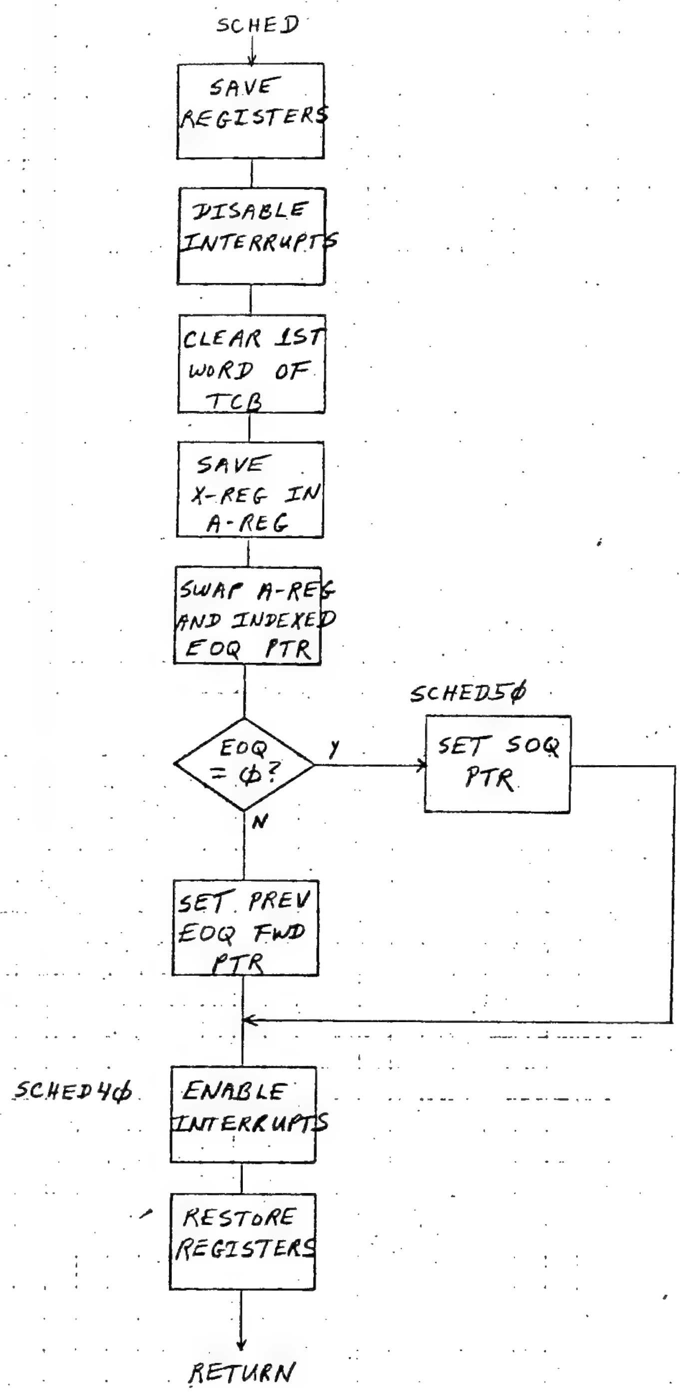
RETURN BLOCK SUBROUTINE

49956 | T. CHERNESKY | 2 JUNE 72

3 | SHEET 1 OF 1

23

FLOW CHART | REMARKS



'TCB' MEANS 'TASK CONTROL BLOCK'

INDEXING IS BY TASK PRIORITY LEVEL

CALL SEQ –

LDSX (ADDR OF TCB)

LDSB (TASK PRIORITY LEVEL)

JSUB (=SCHED)

– NORMAL RETURN

RAYTHEON | RAYTHEON COMPANY LEXINGTON, MASS 02173

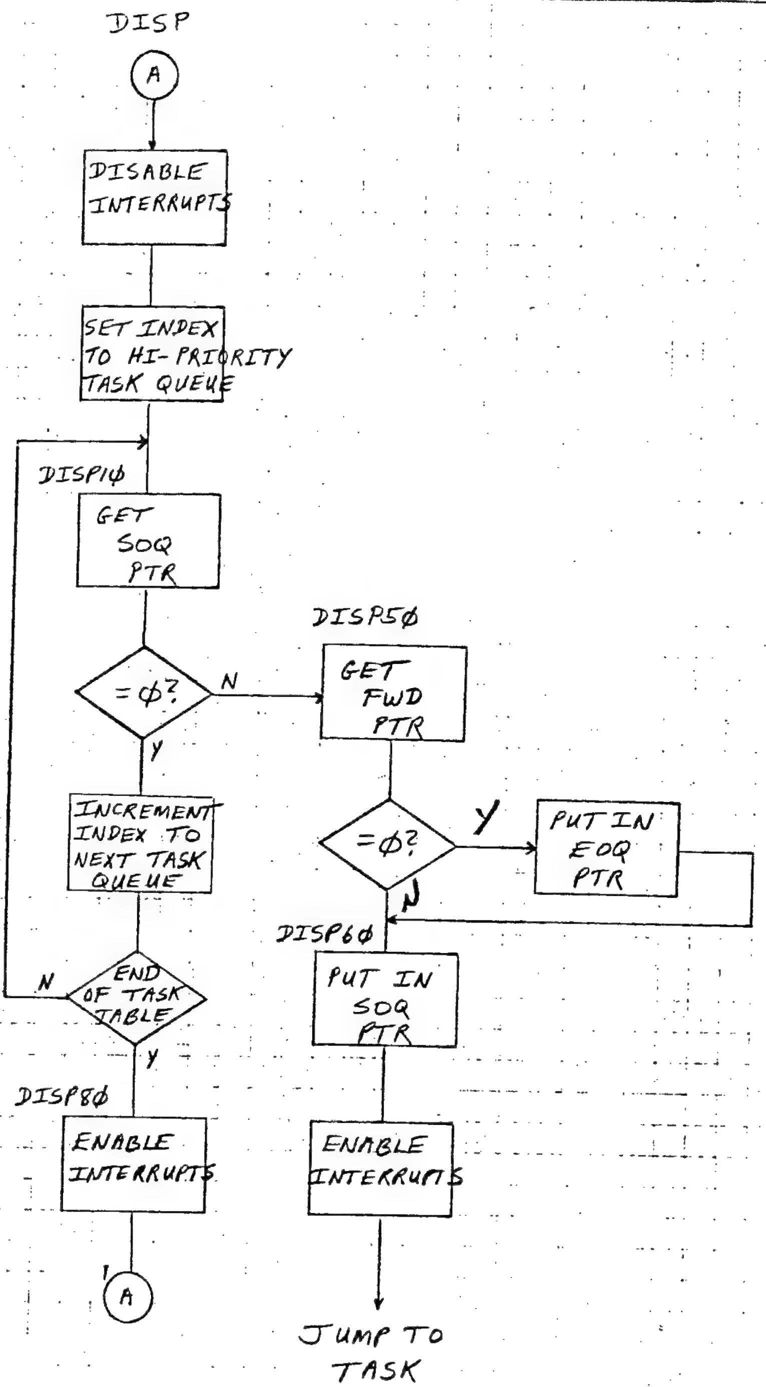
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV

SCHEDULER SUBROUTINE

CODE IDENT NO 49956 | PREPARED BY T. CHERNESKY | DATE 9 APR 76

NUMBER 4 | SHEET 1 OF 1

FLOW CHART | REMARKS



X-REG CONTAINS ADDR OF TASK CONTROL BLOCK

CALL SEQ —

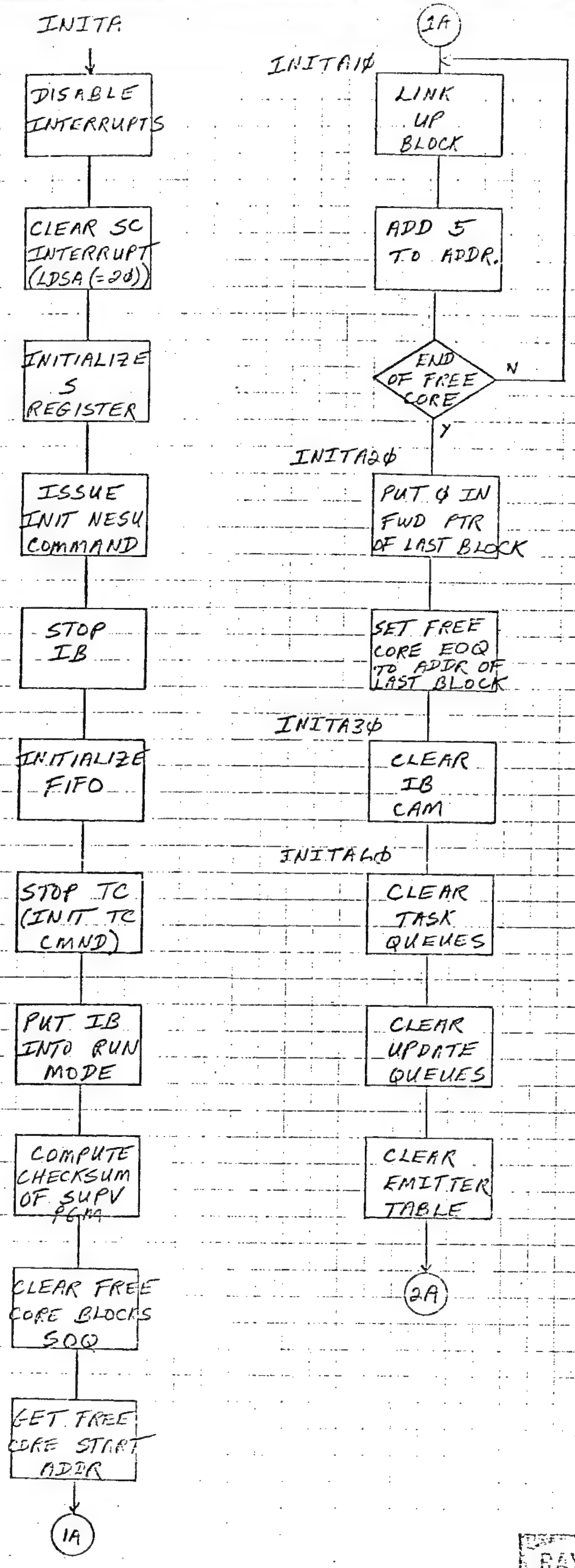
JUMP (=DISP)

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SUPV
DISPATCHER

CODE IDENT NO: 49956
PREPARED BY: T. CHERNESKY
DATE: 9 APR 76

NUMBER: 5 SHEET 1 OF 1



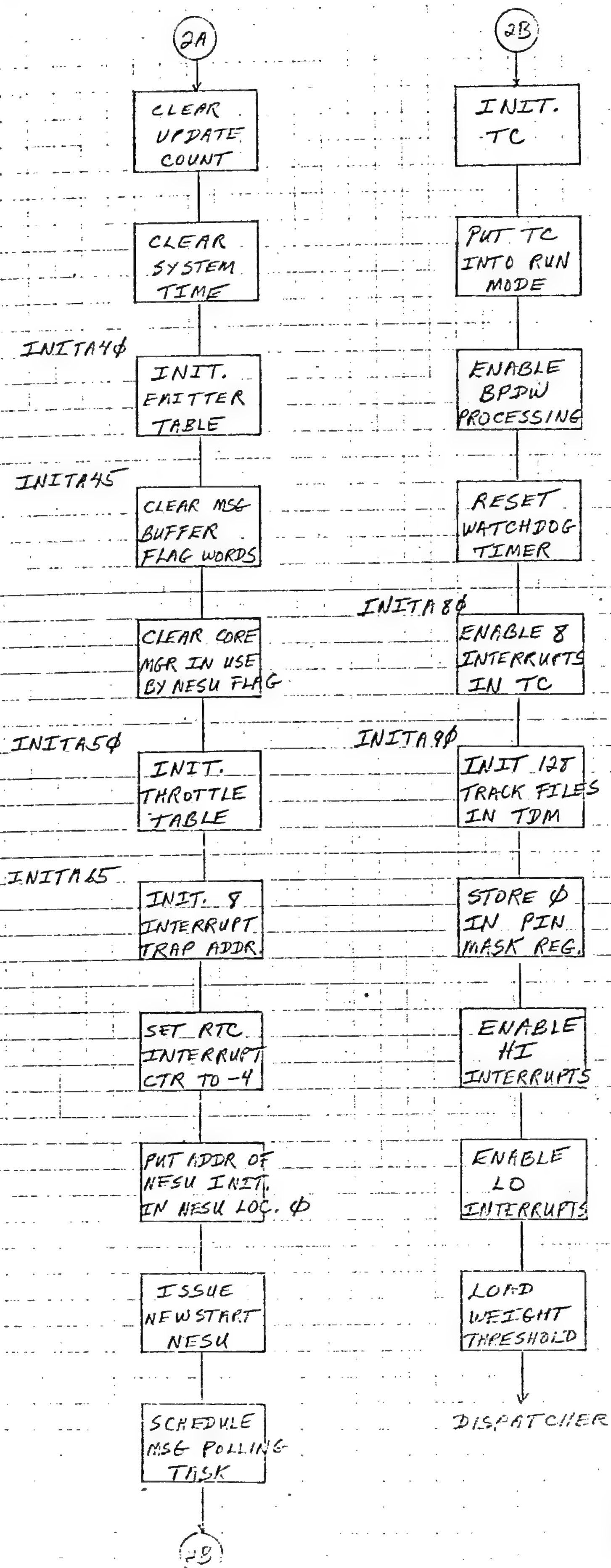
RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

INITIALIZATION MODULE

49956

SHEET 1 OF 2



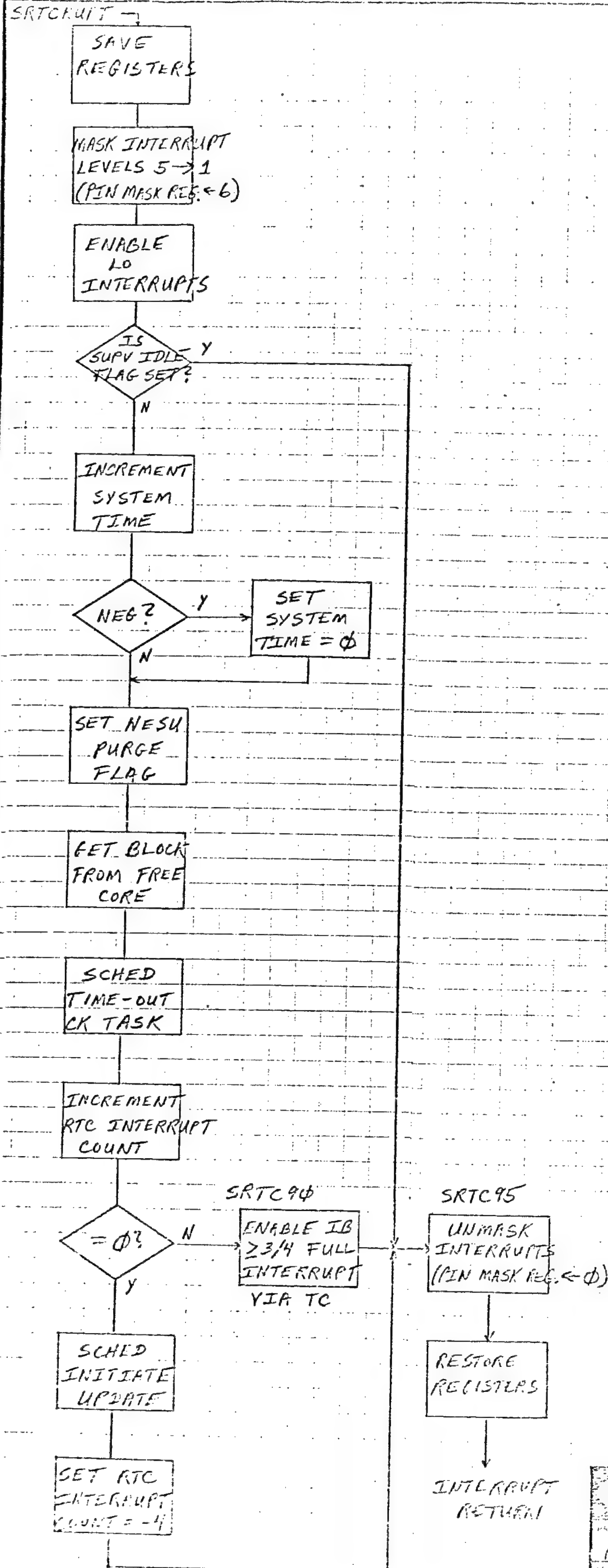
RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS 02173

INITIALIZATION MODULE

49956

7



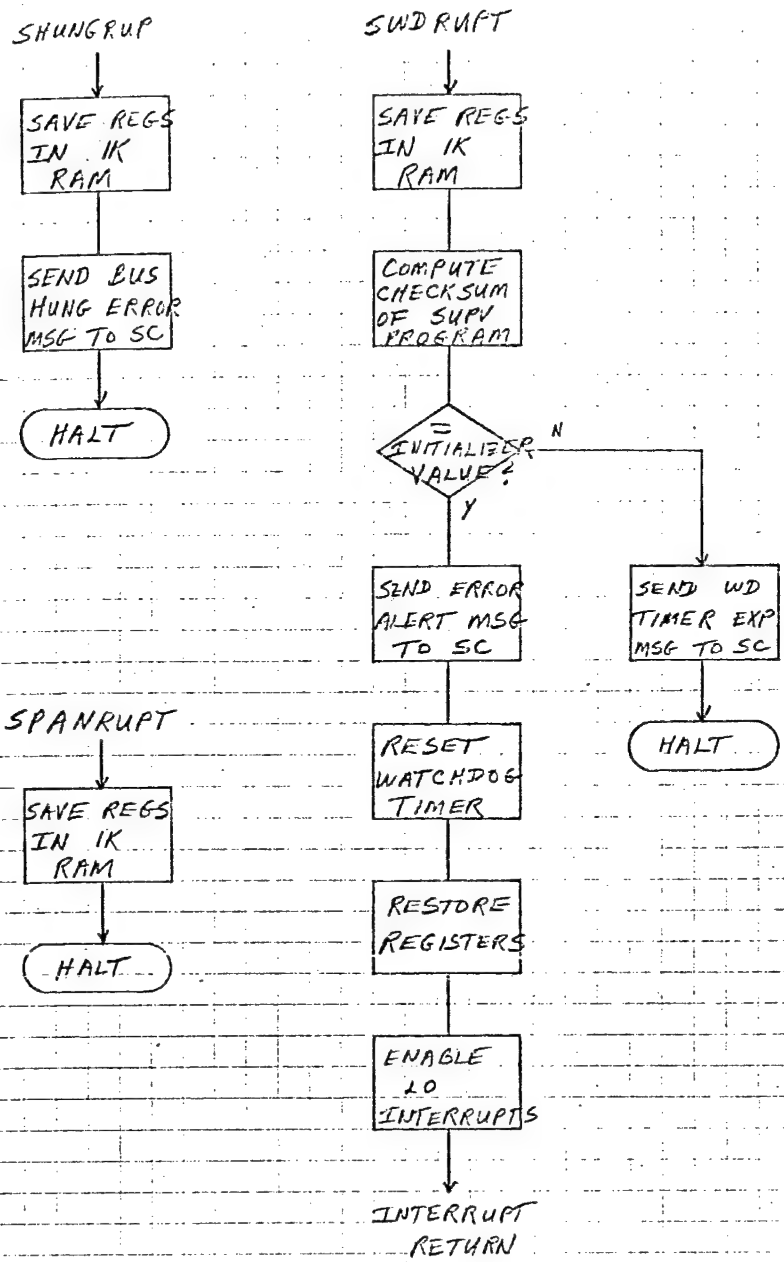
RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS 02173

RTC INTERRUPT HANDLER

49956

1 of 1



REGS. SAVED ARE A, E, B, X, S, P

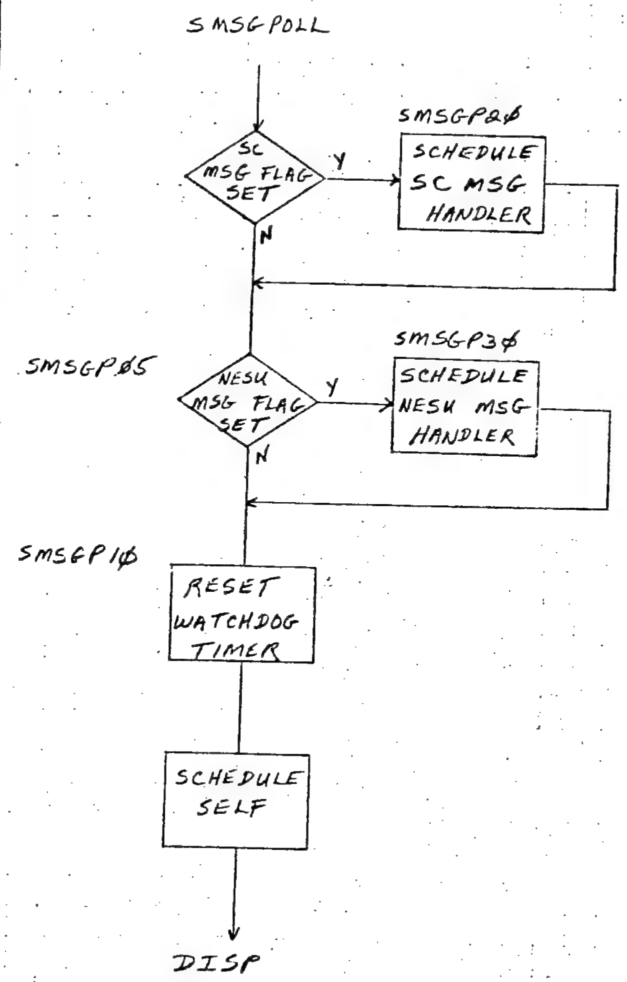
RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

49956

SHEET 1 OF 1

REMARKS



RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

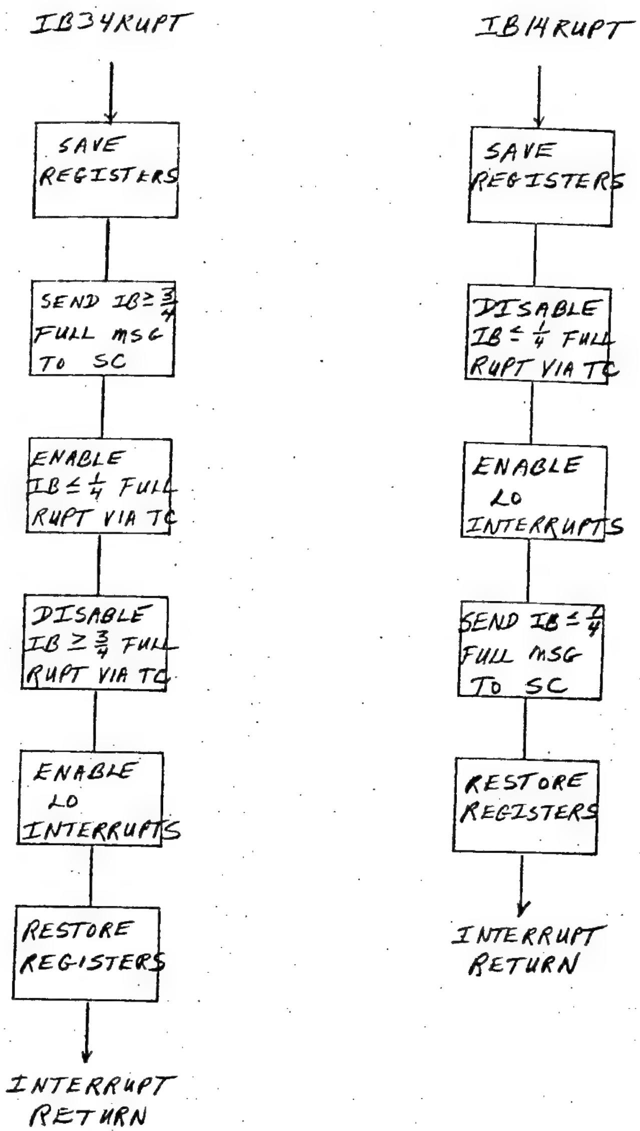
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV
MESSAGE POLLING TASK

CODE IDENT NO	PREPARED BY	DATE
49956	T. CHERNESKY	12 APR 76

NUMBER 23 SHEET 1 OF 1

FLOW CHART

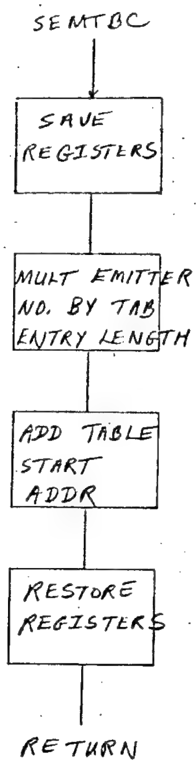
REMARKS



IB $\geq \frac{3}{4}$ FULL INTERRUPT IS ENABLED VIA TC EVERY RTC INTERRUPT

RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV IB INTERRUPT HANDLERS	
CODE IDENT NO 49956	PREPARED BY T. CHERNESKY DATE 30APR76
NUMBER 24	SHEET 1 OF 1

FLOW CHART | REMARKS



CALL SEQ –

LDSA (EMITTER NO.)

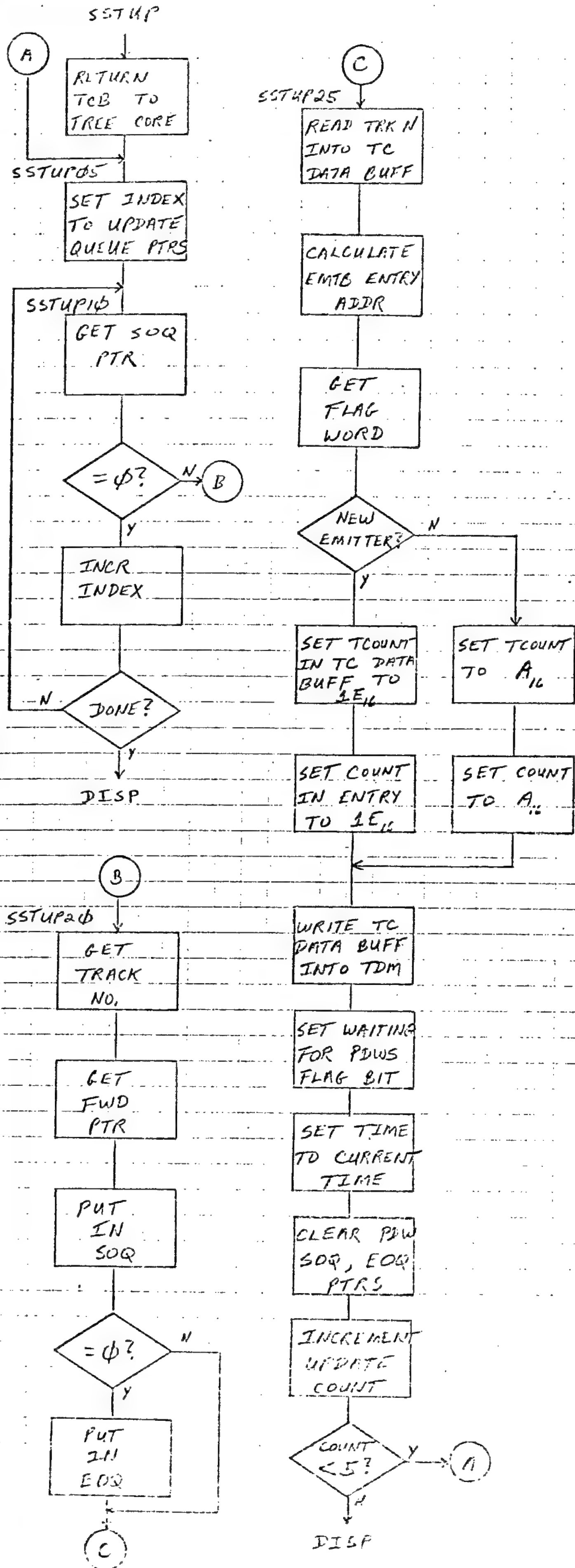
JSUB (= SEMTBC)

STSA (EMTB ENTRY ADDR.)

RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV	
CALC. EMITTER TABLE ENTRY ADDR SUBR.	
CODE IDENT NO 49956	PREPARED BY T. CHERNESKY DATE 30 APR 76
NUMBER 25	SHEET 1 OF 1

10-2319 (8-72)

FLOW CHART | REMARKS

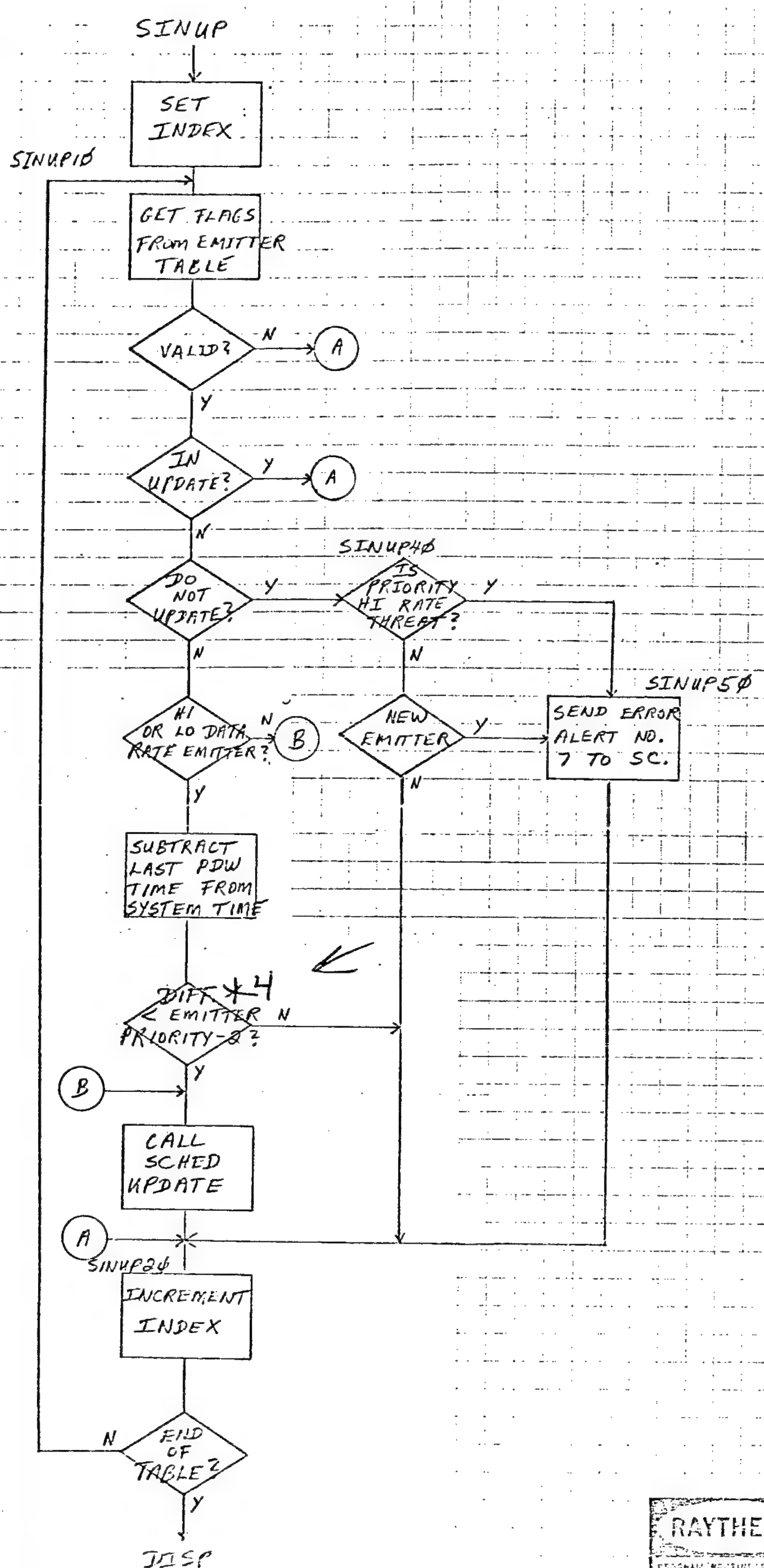


RAYTHEON | RAYTHEON COMPANY
LEXINGTON, MASS 02173

START UPDATE TASK

49956

SHEET 1 OF 1



EMITTER UPDATE INTERVALS

1 SEC
- NEW EMITTER
- HI RATE THREAT

2 SEC
- HI RATE

4 SEC
- LO RATE

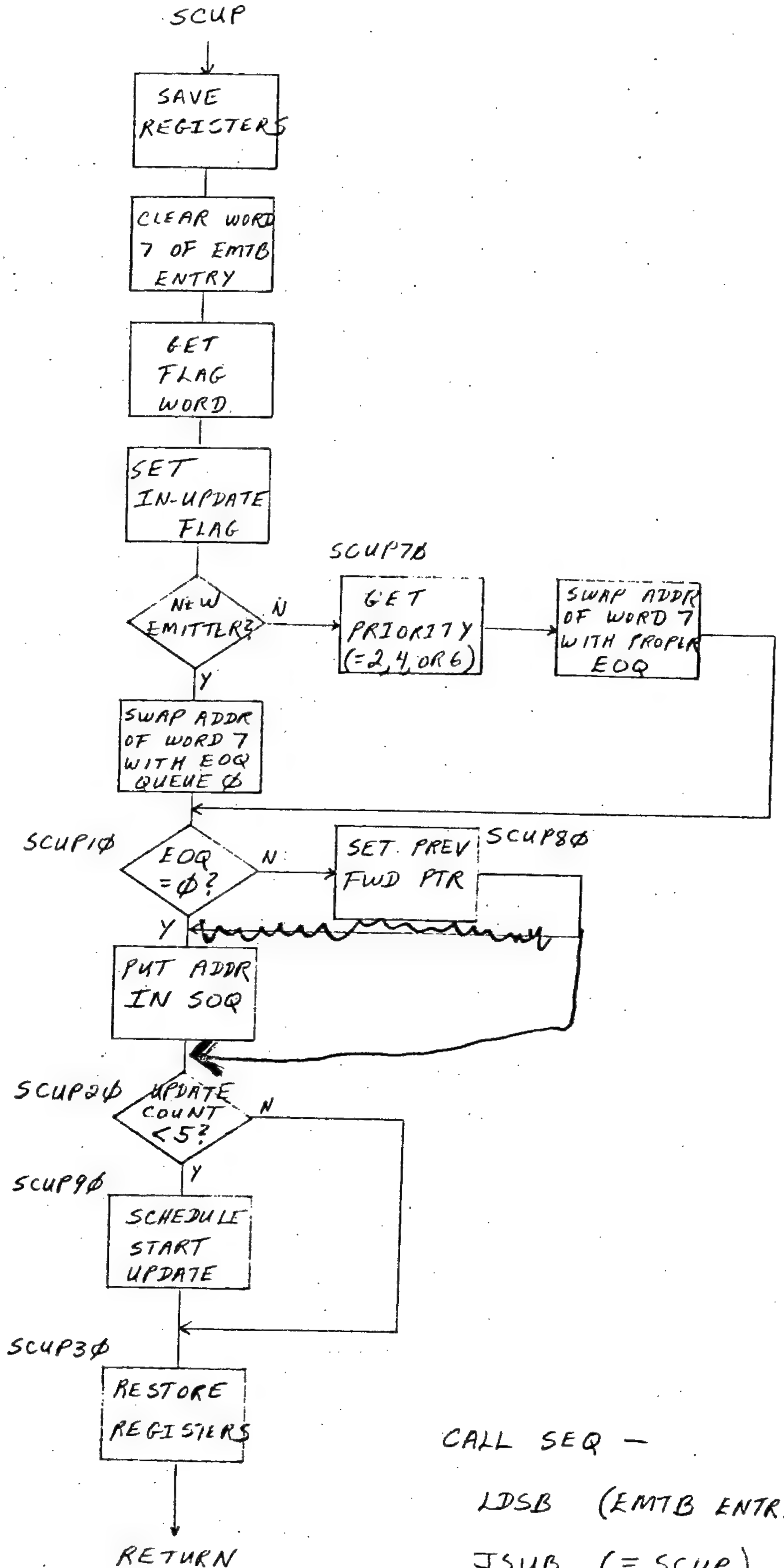
RAYTHEON COMPANY
LEXINGTON, MASS. 02173

IEWS SORTER SURV
INITIATE UPDATE TASK

49956 | T. C... | 1 JUN 76

NUMBER 9 | SHEET 1 OF 1

FLOW CHART	REMARKS



'EMTB' MEANS 'SUPV EMITTER TABLE'

CALL SEQ —

LDSB (EMTB ENTRY ADDR)

JSUB (=SCUP)

— NORMAL RETURN

RAYTHEON

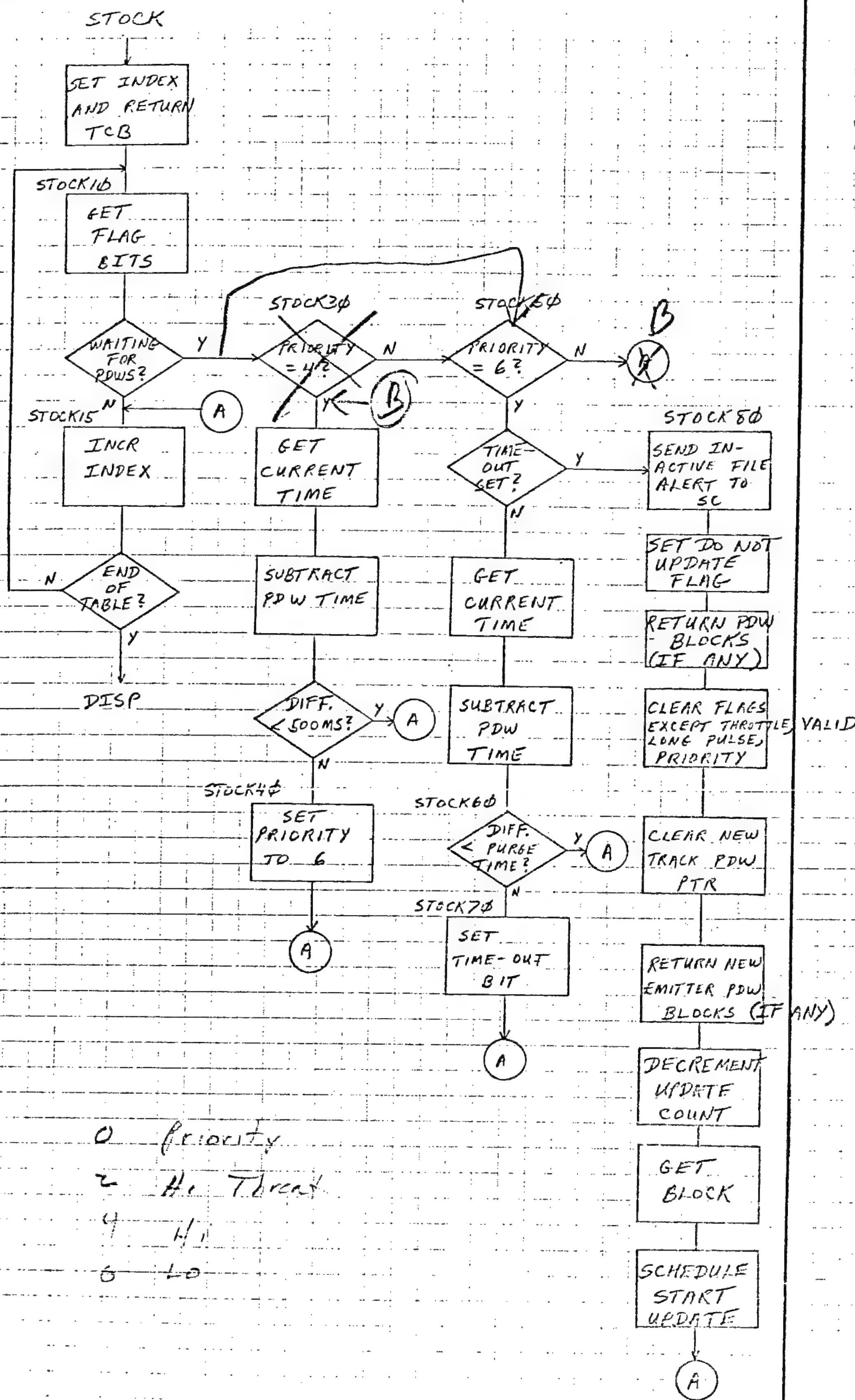
RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV
SCHEDULE UPDATE SUBROUTINE

CODE IDENT NO 49956 | PREPARED BY T. CHERNESKY | DATE 12 APR 76

NUMBER 1Ø | SHEET 1 OF 1

FLOW CHART | REMARKS



RAYTHEON

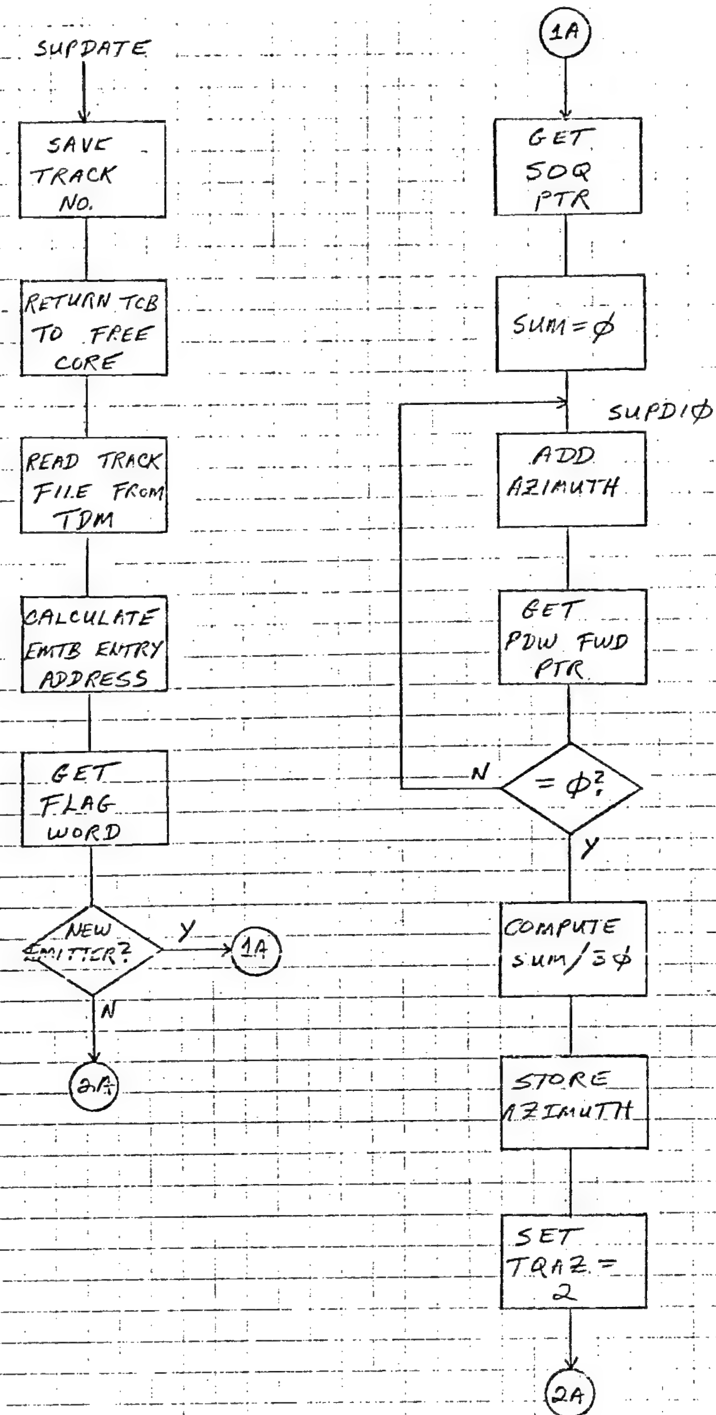
RAYTHEON COMPANY
LEXINGTON, MASS. 02173

TIME OUT CHECK TASK

49956 | [illegible] | 1 JUN 7[illegible]

SHEET 1 OF 1

FLOW CHART | REMARKS



AZIMUTH CALCULATION

RAYTHEON

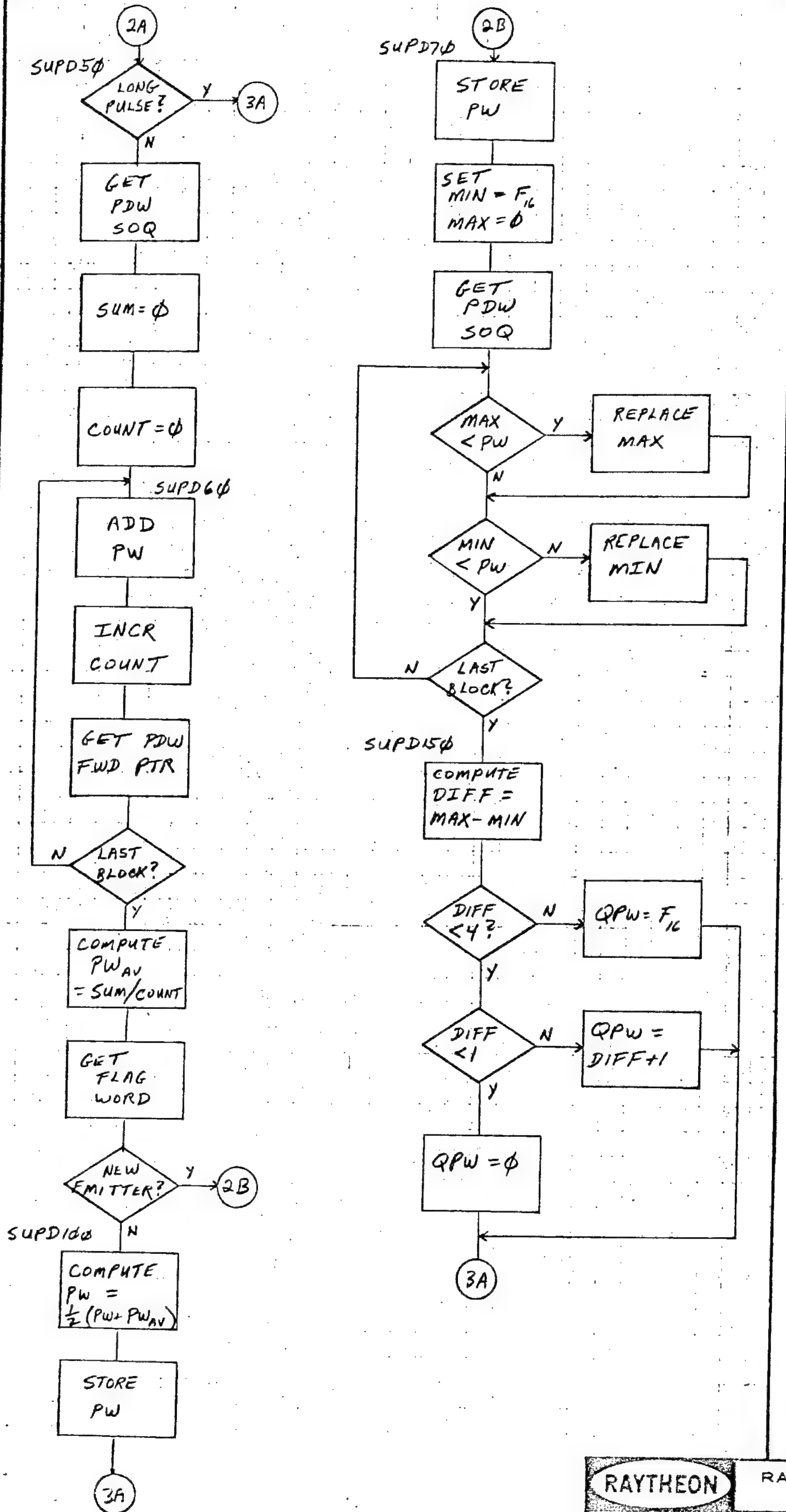
RAYTHEON COMPANY
LEXINGTON, MASS 02173

UPDATE TASK

49956 | 3 JUN 76

2 | SHEET 1 OF 9

FLOW CHART



REMARKS

PULSE WIDTH CALCULATION

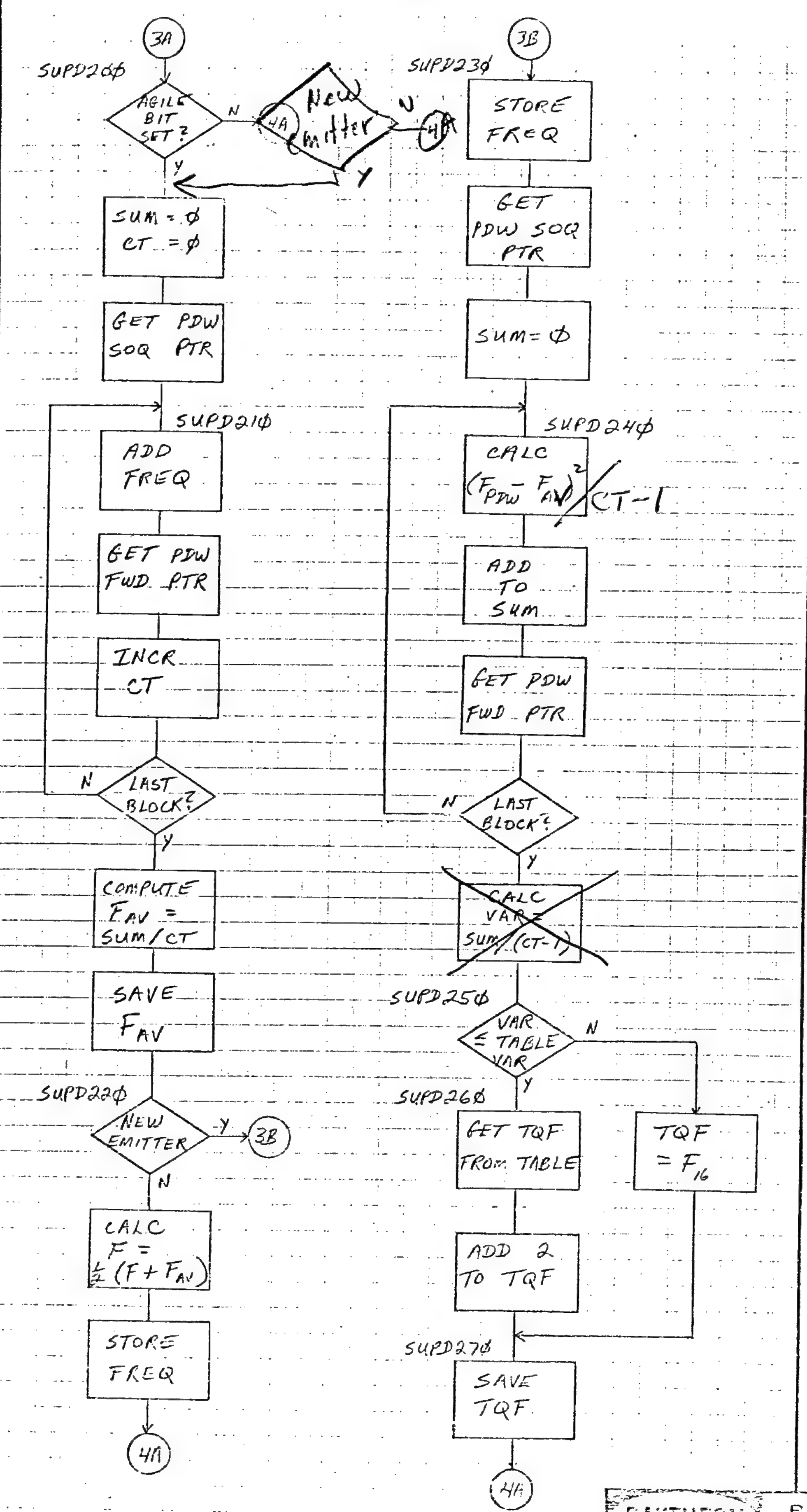
RAYTHEON — RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SUPV UPDATE TASK

CODE IDENT NO	PREPARED BY	DATE
49956	T. CHERNESKY	5 MAY 76

NUMBER 13 | SHEET 2 OF 9

FLOW CHART | REMARKS



FREQUENCY CALCULATION

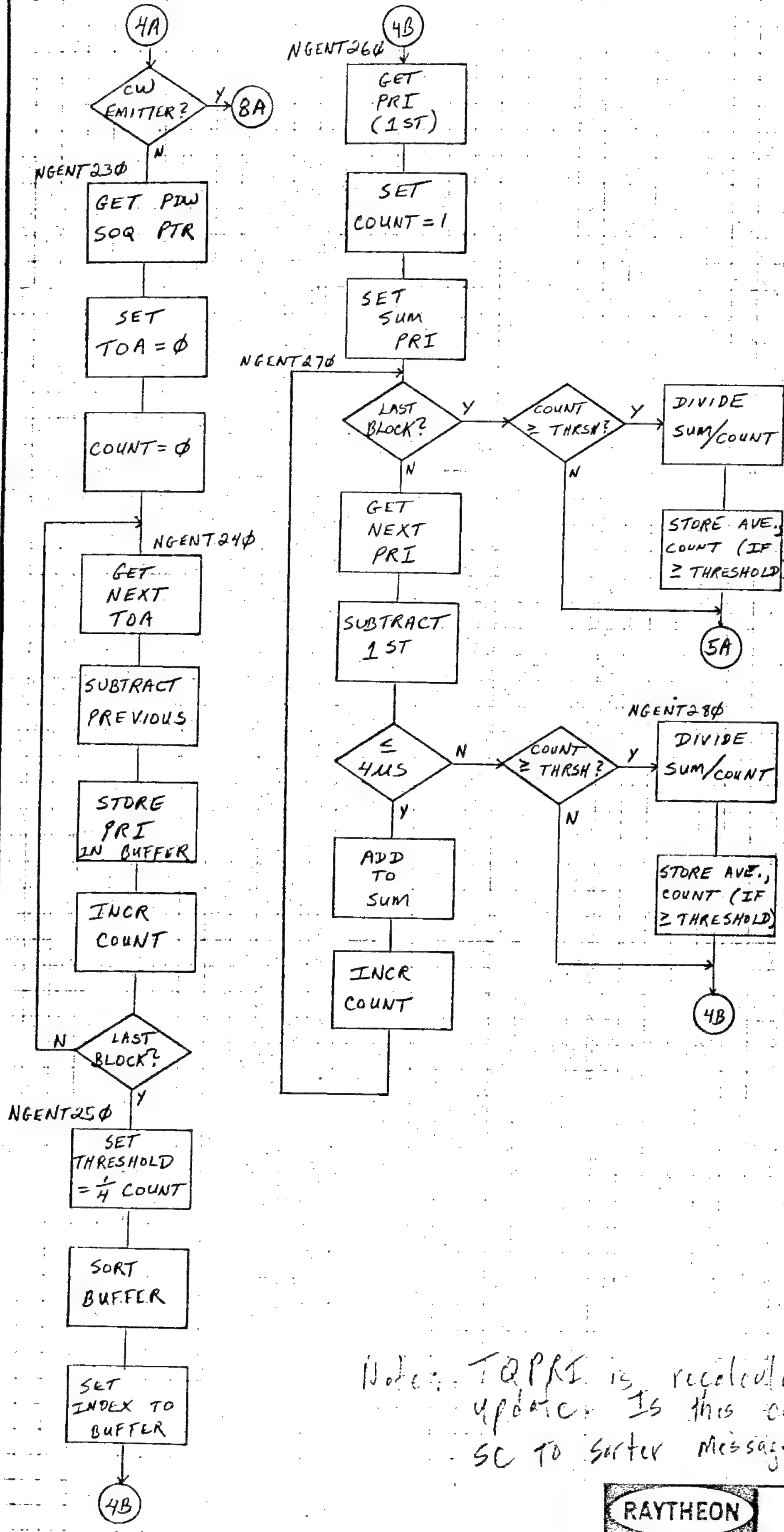
Calculation of one term of the variance estimate.

RAYTHEON | RAYTHEON COMPANY
LEXINGTON, MASS 02173

IEWS SORTER SUPV
UPDATE TASK
49956

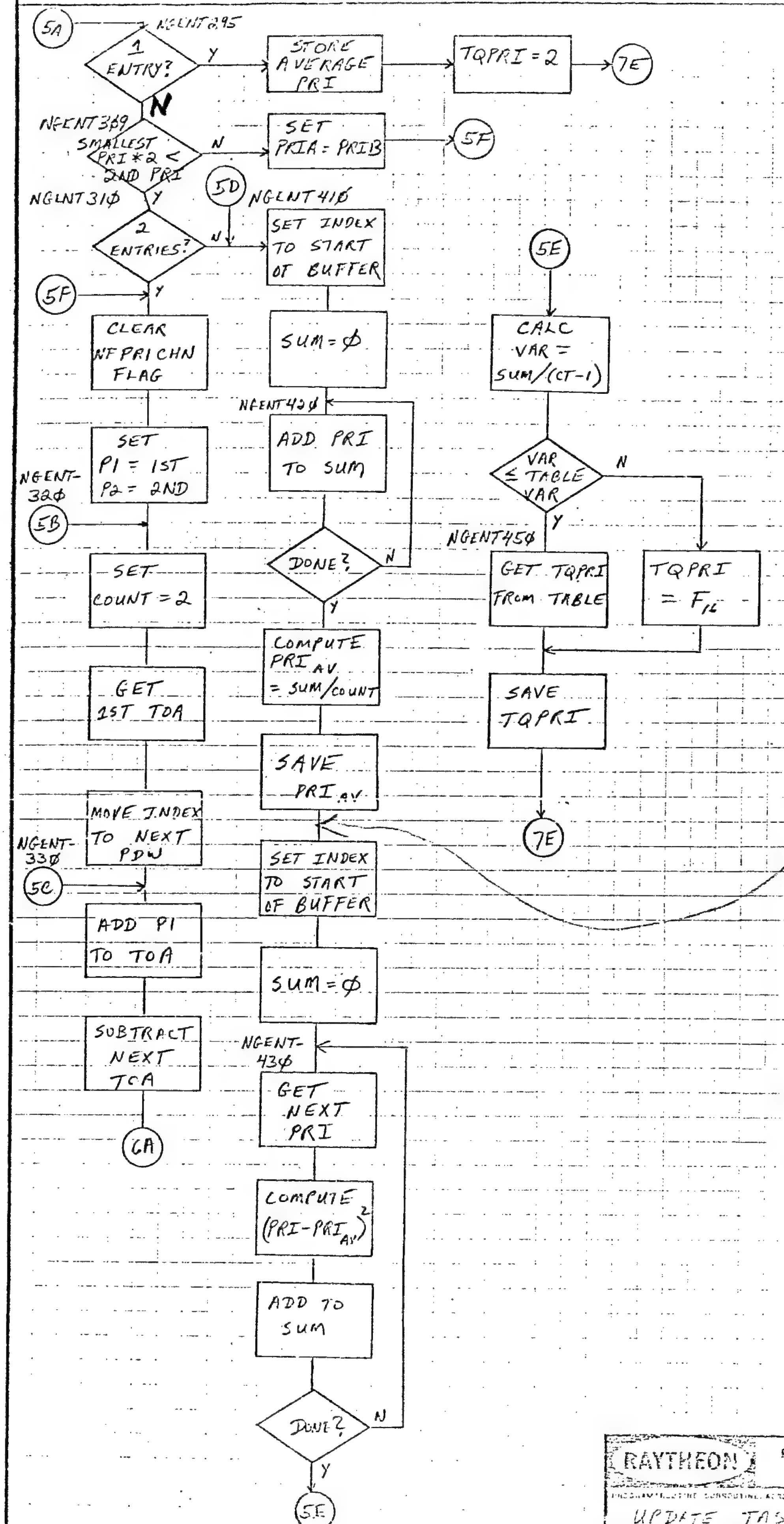
PRI CALCULATION

Find no. of groups of PRI within 4μs and count in group ≥ threshold



Note: TQPRI is recalculated every update. Is this consistent with SC to Sorter Message? No, set TQPRI only if new emitter

RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM	IEWS SORTER SUPV UPDATE TASK
CODE IDENT NO 49956	PREPARED BY T. CHERNESKY DATE 5 MAY 76
NUMBER 15	SHEET 4 OF 9



PRI CALCULATION

CHAINING AND VARIANCE CALCULATION

should skip variance calc if not new emitter

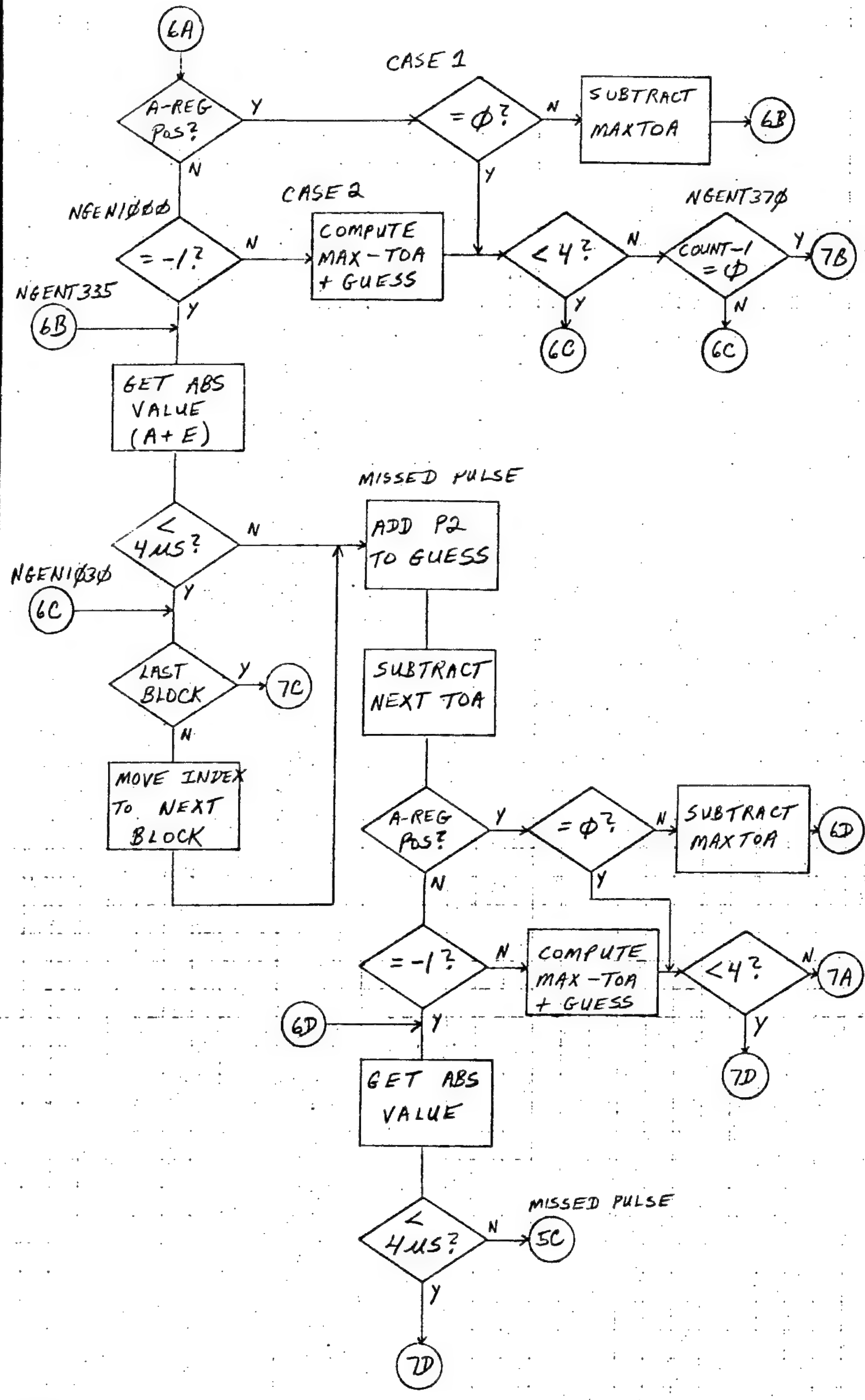
RAYTHEON | RAYTHEON COMPANY
LEXINGTON, MASS 02173

ILWS SCALER SURV
UPDATE TASK

49956 | T. CHERNESKY | 3 JUN 76

SHEET 5 OF 9

FLOW CHART | REMARKS



PRI CHAIN CALCULATION – TOA WRAPAROUND PROBLEM

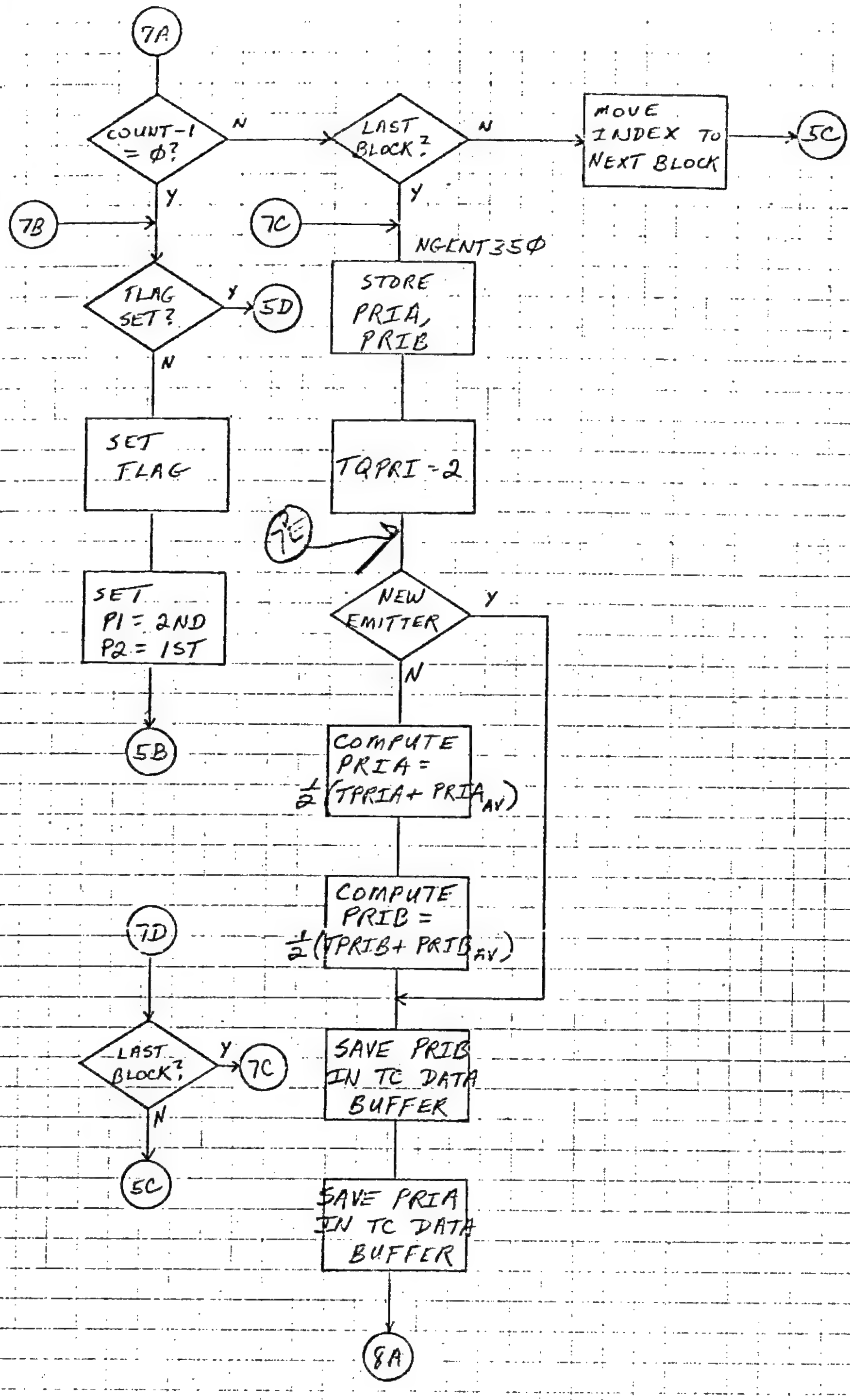
CASE 1 : GUESS = MAX–N $N = \emptyset, 1, 2, 3$
ACTUAL = M $M = 3, 2, 1, \emptyset$

GUESS – ACTUAL = LARGE POS NO., I.E., $MAX-N-M > 2^{16}$
THEREFORE, NEED $N+M < 4$, ELSE POSSIBLE MISSED PULSE.
IF A-REG POS AND NON-ZERO, THEN COMP((GUESS–ACT)–MAX) = N+M

CASE 2 : GUESS = M $M = \emptyset, 1, 2, 3$
ACTUAL = MAX – N $N = 3, 2, 1, \emptyset$
GUESS – ACTUAL = LARGE NEG. NO., M – MAX – N
THEREFORE, $N+M < 4$, ELSE ERROR.
IF A NEG AND $A \neq -1$, THEN
MAX – ACT + GUESS = MAX – (MAX – N) + M = N+M

RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM	IEWS SORTER SUPV UPDATE TASK
CODE IDENT NO 49956	PREPARED BY T. CHERNESKY	DATE 7 MAY 76
NUMBER 17	SHEET 6 OF 9

FLOW CHART | REMARKS



PRI CHAIN CALCULATION

RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS 02173

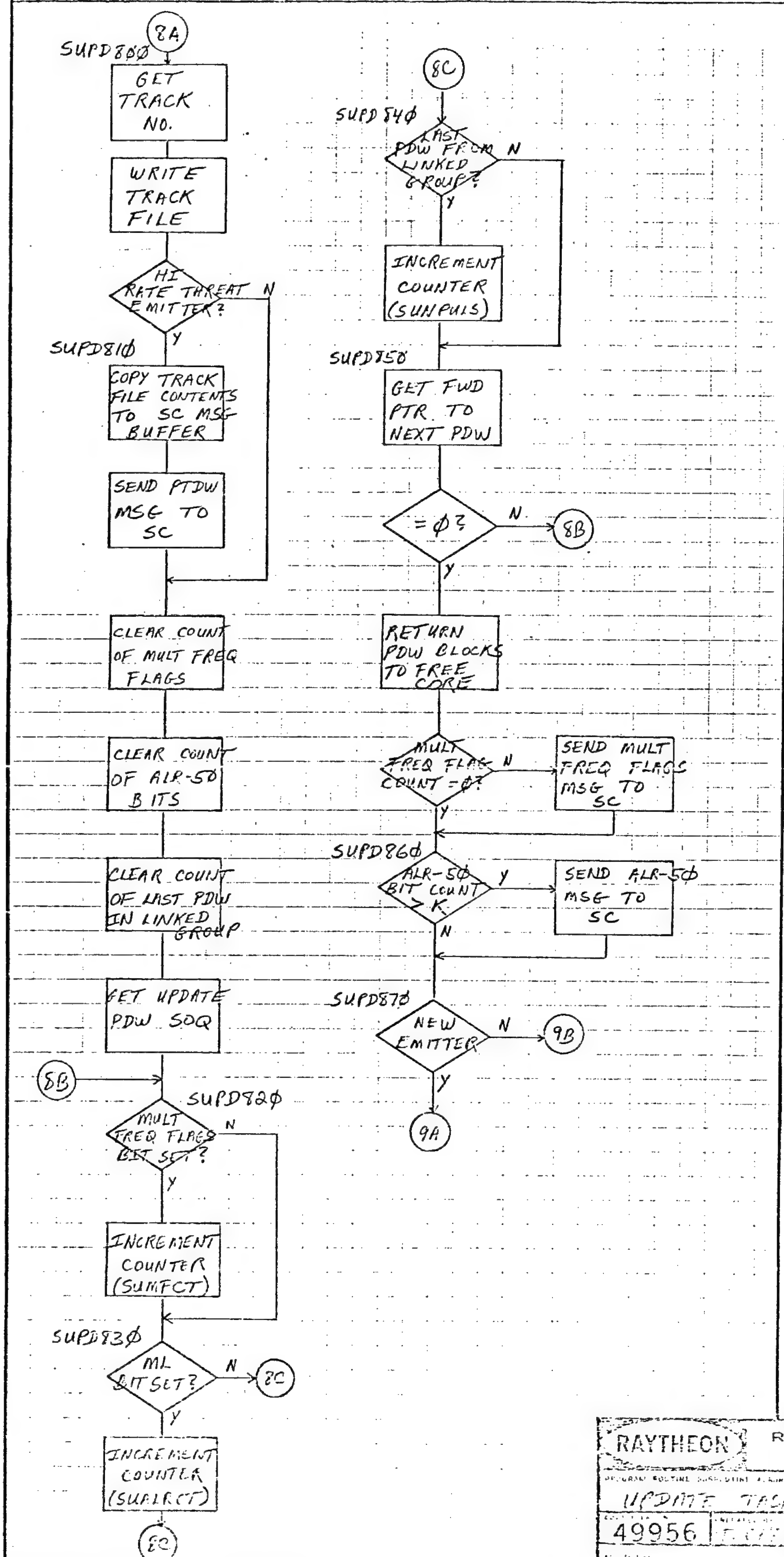
PROGRAM ROUTINE SUBROUTINE ACRONYM: IEWS SORTER SUPV

UPDATE TASK

49956 | T. CHERNESKY | DATE: 3 JUN 76

NUMBER: 12 | SHEET 7 OF 7

FLOW CHART



REMARKS

WRITE TRACK FILE AND OUTPUT MESSAGES TO SC

RAYTHEON

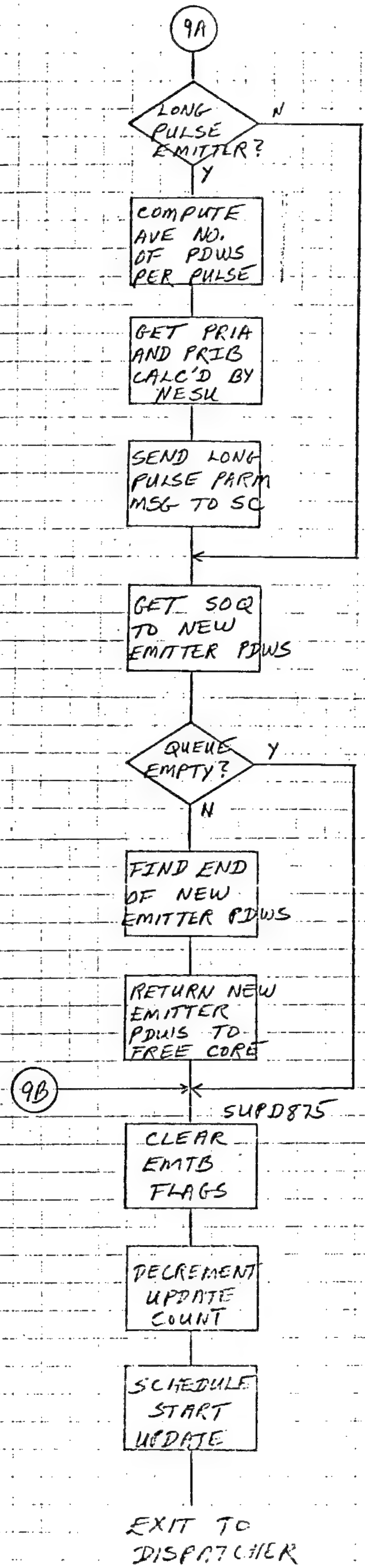
RAYTHEON COMPANY
LEXINGTON MASS 02173

IEWS SORTER SUPV
UPDATE TASK

49956

3 JUN 76

44



WRITE TRACK FILE AND OUTPUT MESSAGES TO SC

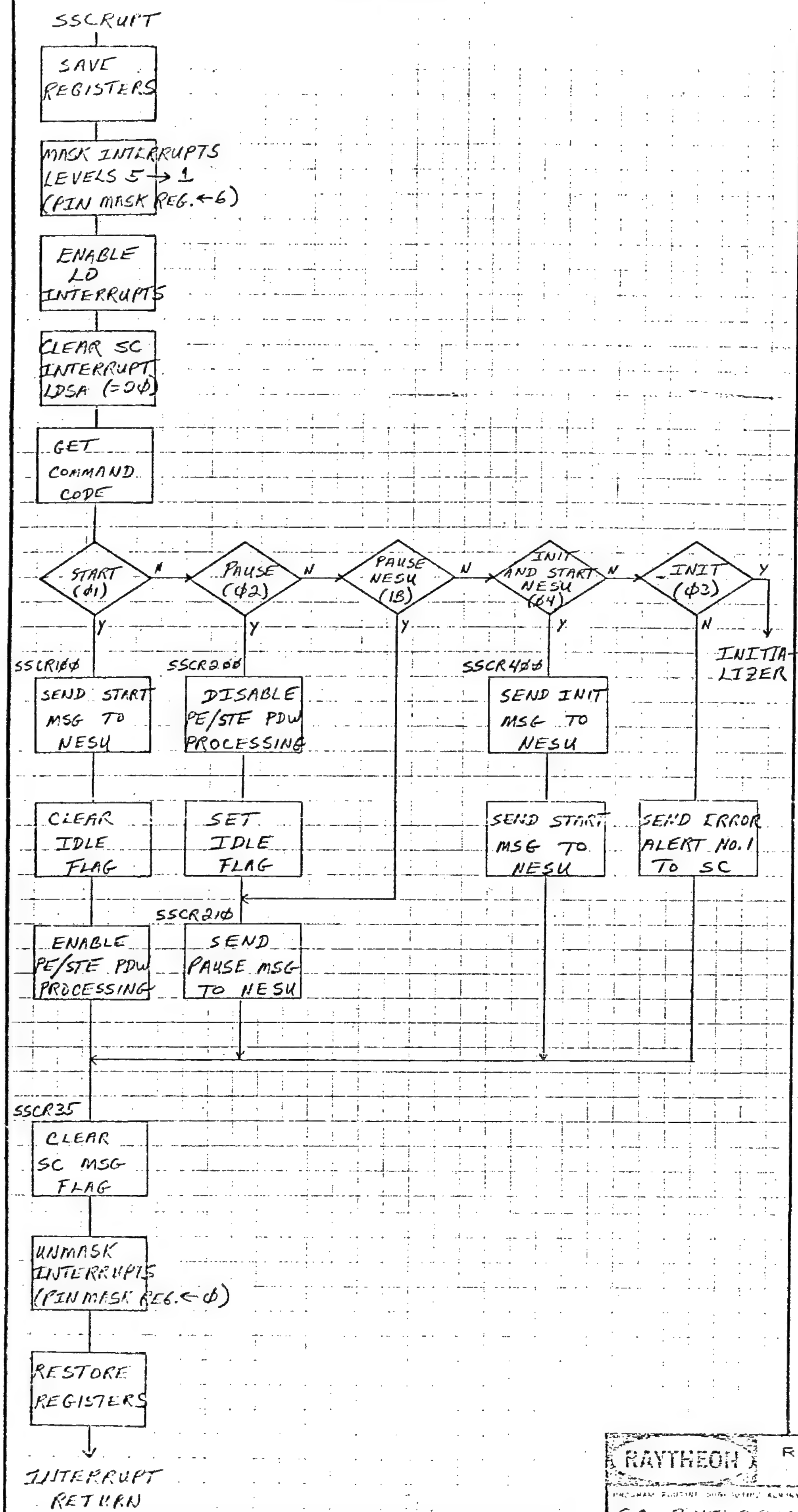
FLAGS CLEARED:
1. IN UPDATE
2. WAITING FOR PDWS
3. TIME OUT
4. NEW EMITTER

RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE ACRONYM: IEWS SORTER SUPV
UPDATE TASK

49956	T. CHERNESKY	3 JUN 76
NUMBER 20	SHEET 9 OF 9	



RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

NEVIS SORTER SHFV

SC INTERRUPT HANDLER

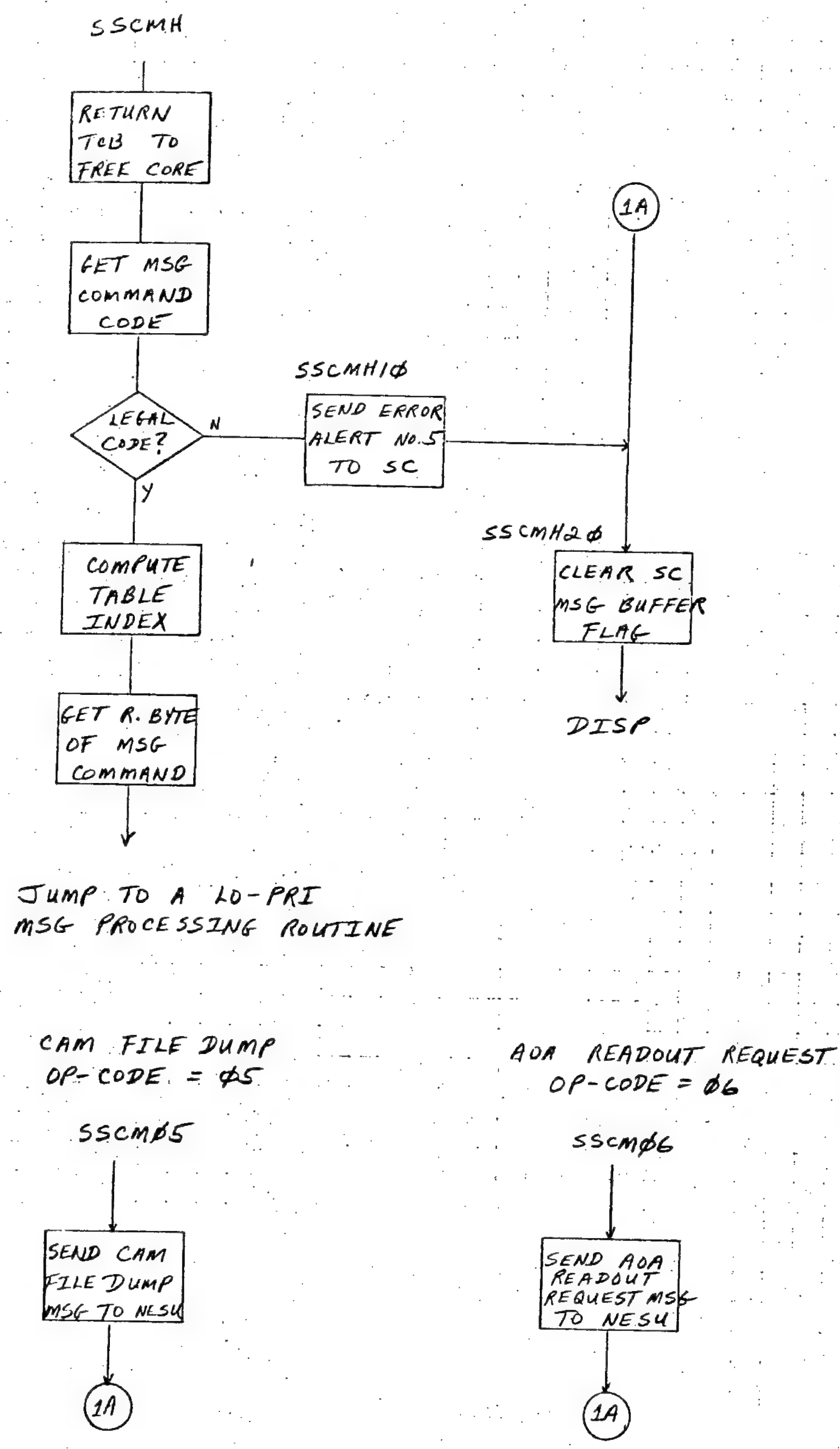
49955 | T. CHERNESKY | 2 JAN 76

SHEET 1 OF 1

46

FLOW CHART

REMARKS



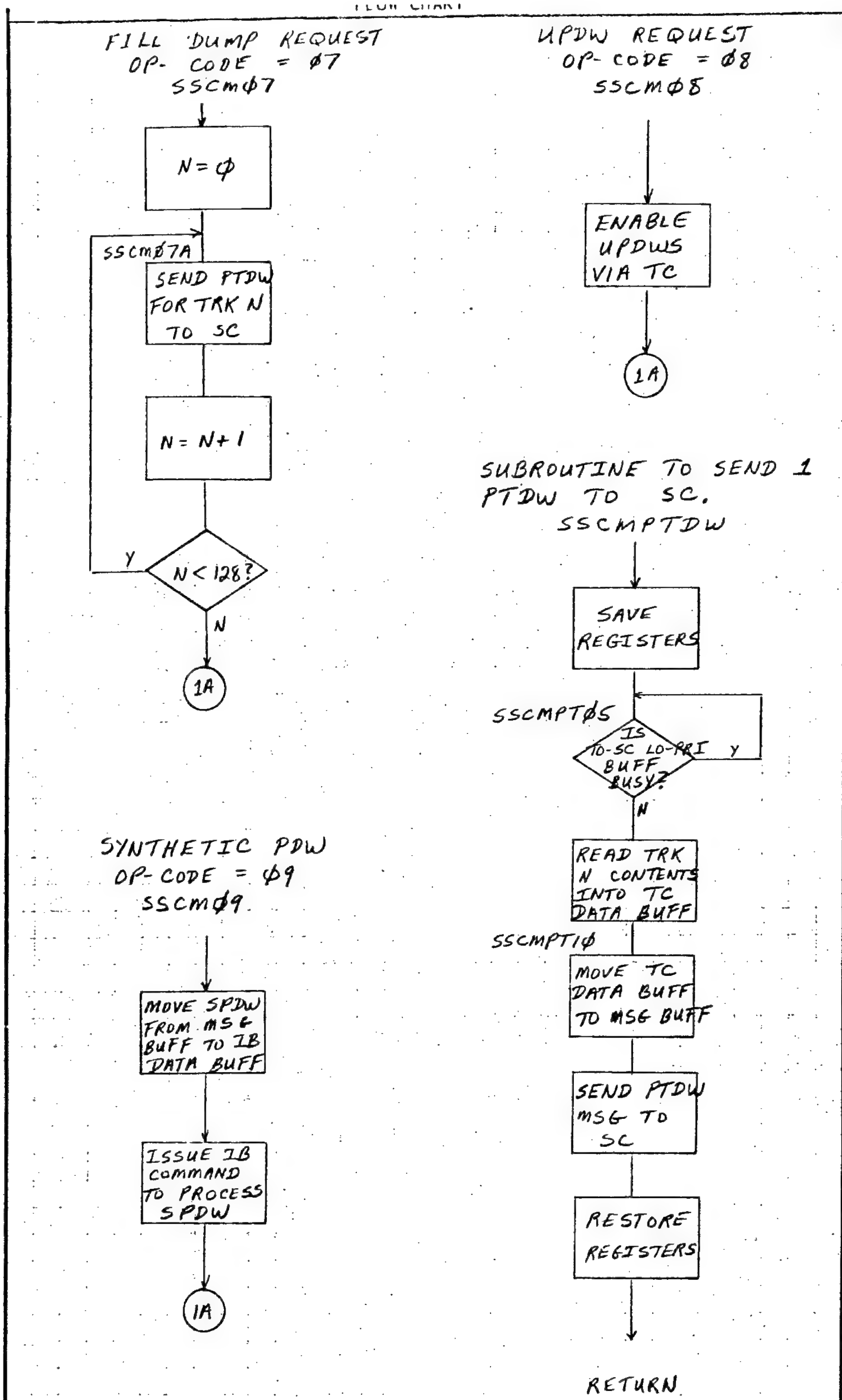
RIGHT BYTE USUALLY CONTAINS SORTER TRACK FILE NO.

RAYTHEON | RAYTHEON COMPANY LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV SC MESSAGE HANDLING

CODE IDENT NO 49956 | PREPARED BY T. CHERNESKY | DATE 12 APR 76

NUMBER 32 | SHEET 1 OF 8

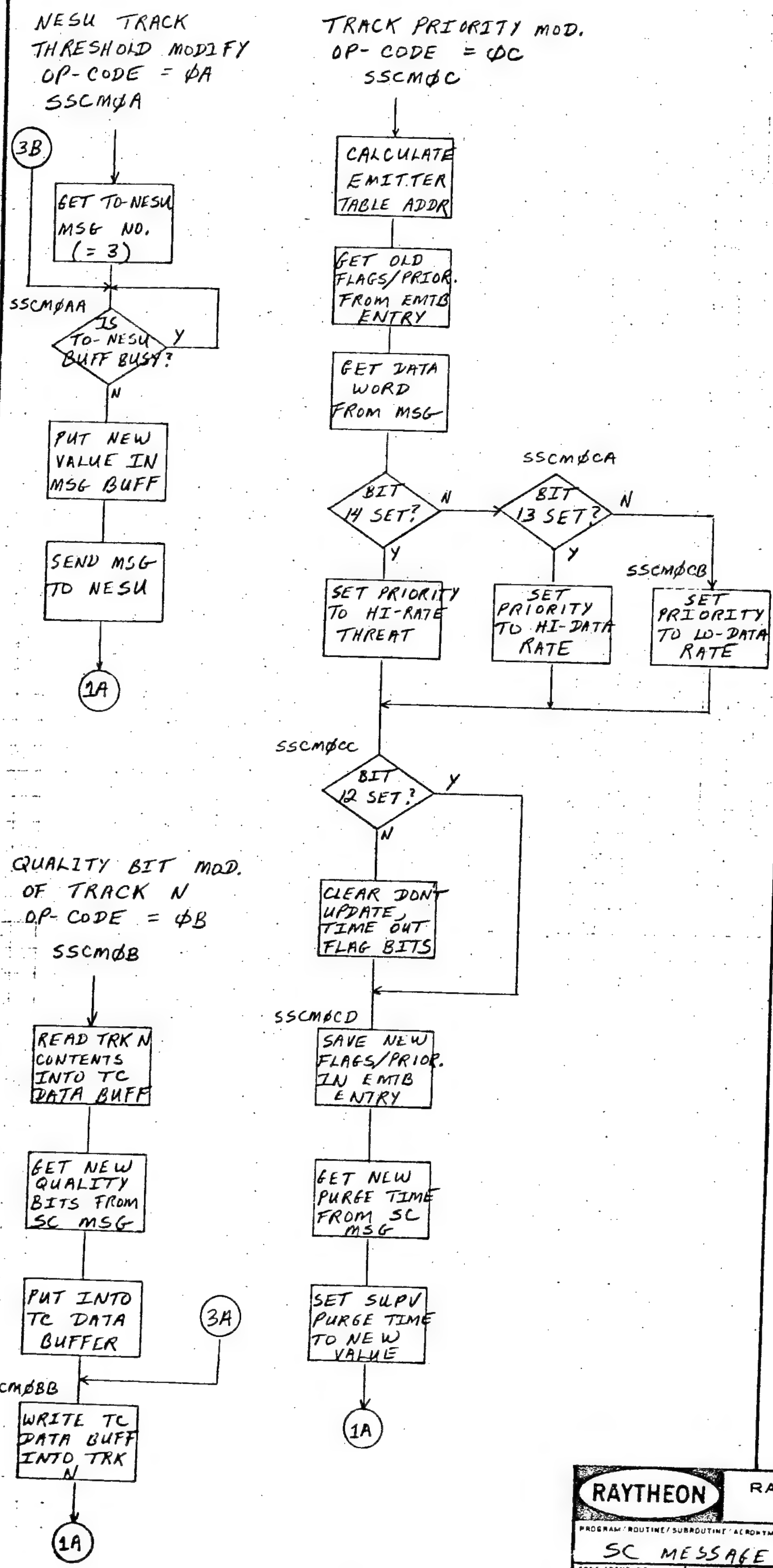


RAYTHEON | RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SURV
SC MESSAGE HANDLING

CODE IDENT NO: 49956 | PREPARED BY: T. CHERNESKY | DATE: 12 APR 76

NUMBER: 33 | SHEET 2 OF 8



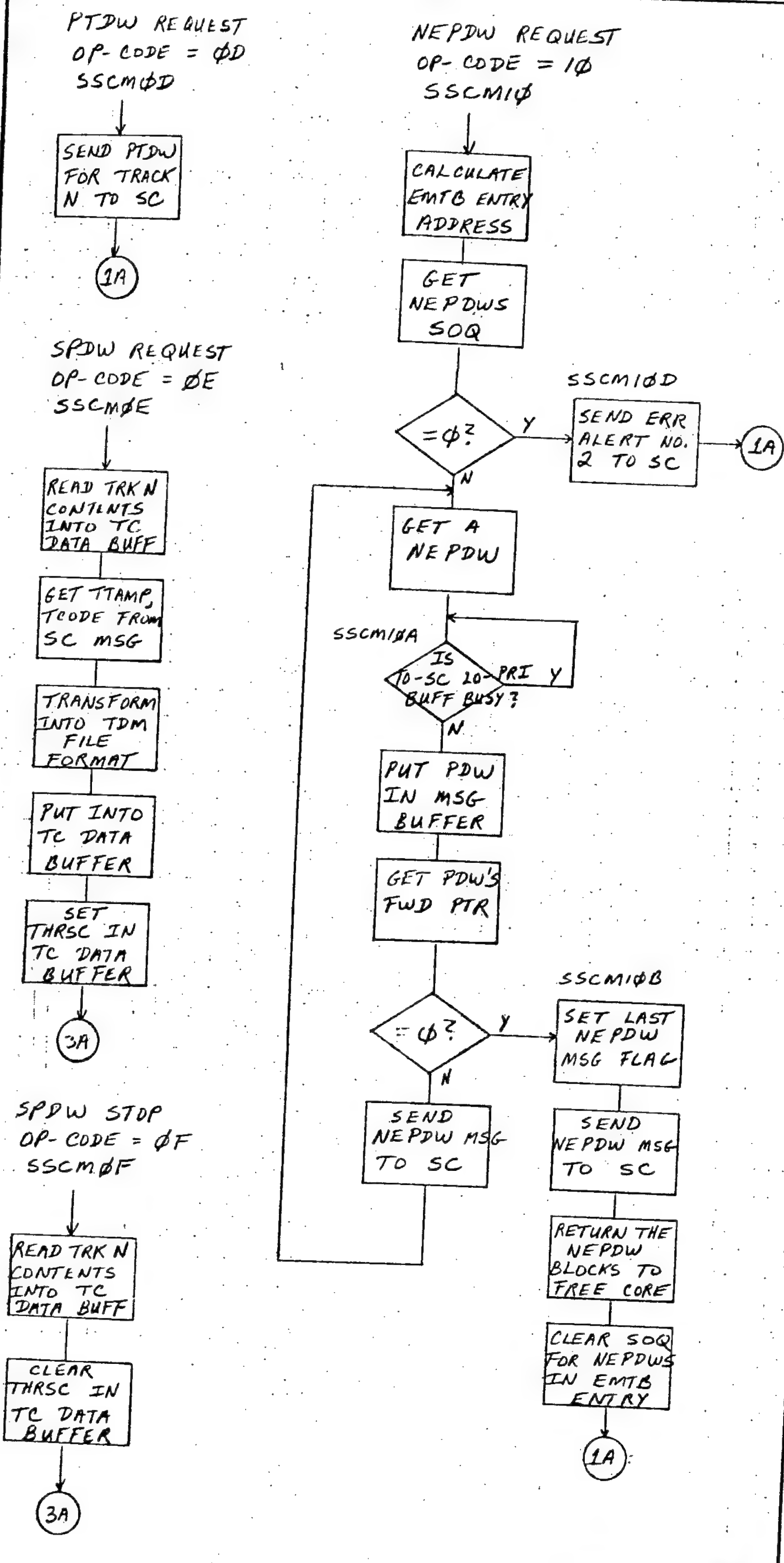
RAYTHEON | RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SUPV
SC MESSAGE HANDLING

CODE IDENT NO: 49956 | PREPARED BY: T. CHERNESKY | DATE: 12 APR 76

NUMBER: 34 | SHEET 3 OF 8

FLOW CHART



REMARKS

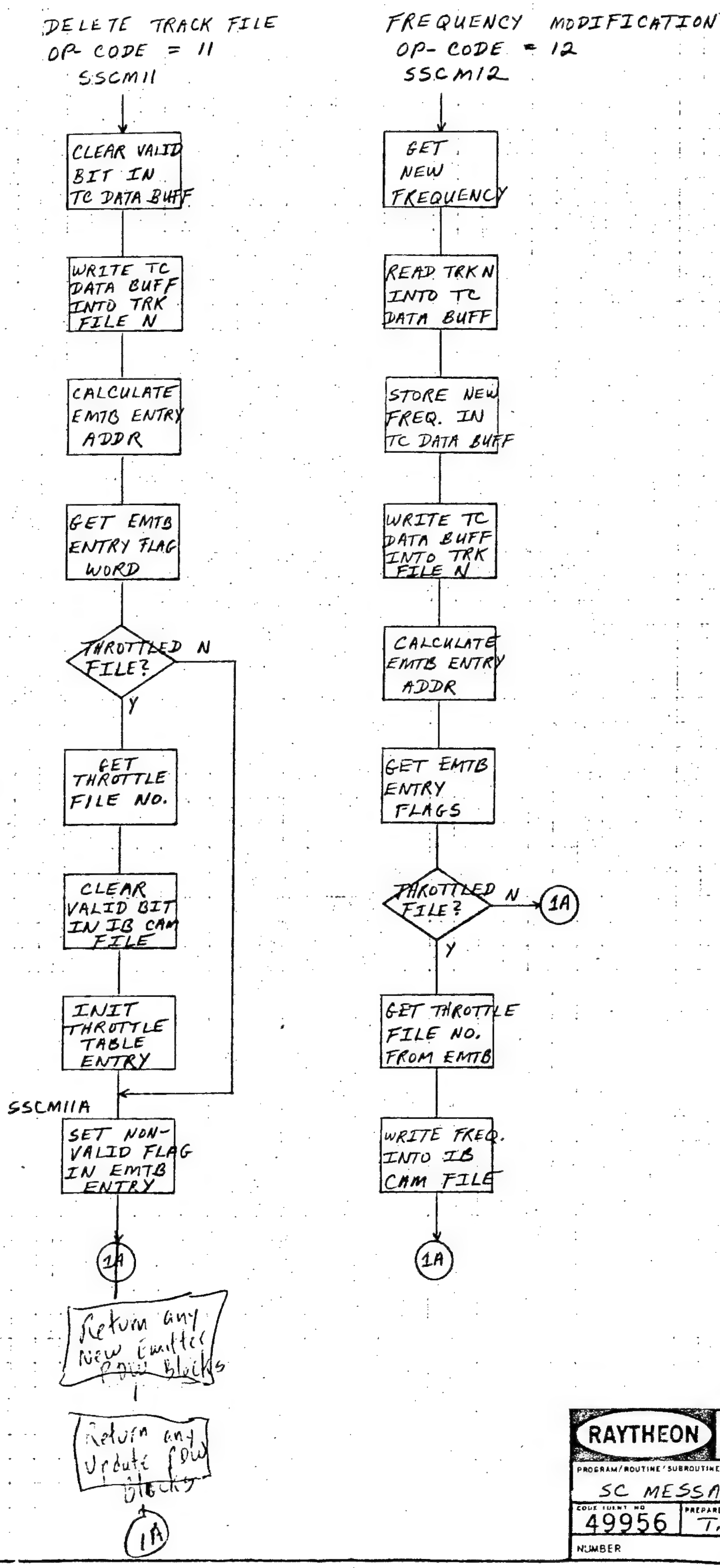
'NEPDW' MEANS 'NEW EMITTER PDW'

RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SUPV
SC MESSAGE HANDLING

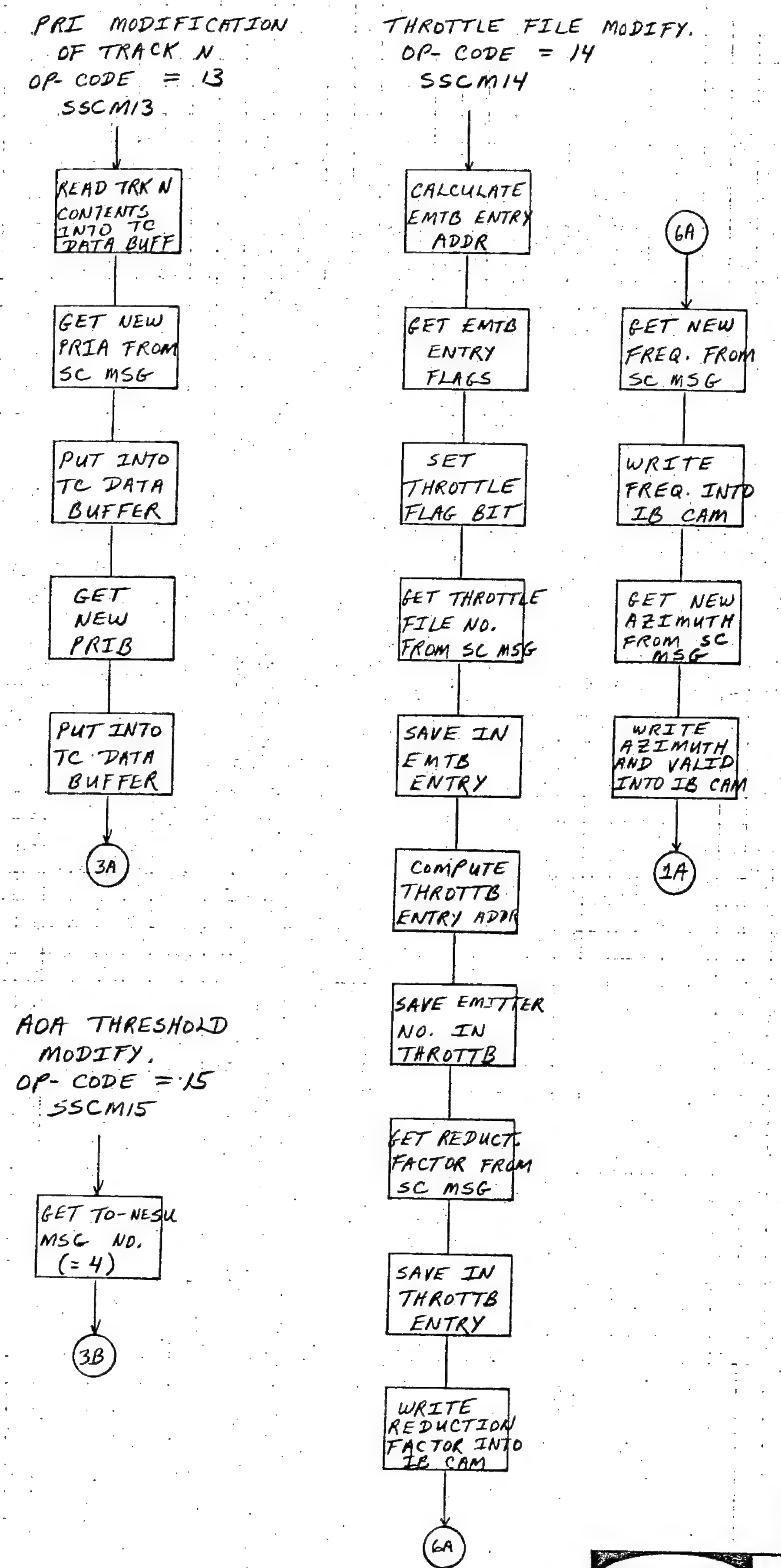
CODE IDENT NO: 49956
PREPARED BY: T. CHERNESKY
DATE: 12 APR 76
NUMBER: 25
SHEET 4 OF 8



RAYTHEON	RAYTHEON COMPANY LEXINGTON, MASS. 02173
PROGRAM/ROUTINE/SUBROUTINE/ACRONYM	IEWS SORTER SUPV SC MESSAGE HANDLING
CODE IDENT NO 49956	PREPARED BY T. CHERNESKY · DATE 12 APR 76
NUMBER 36	SHEET 5 OF 8

SC-2319 (8 72)

FLOW CHART | REMARKS



RAYTHEON

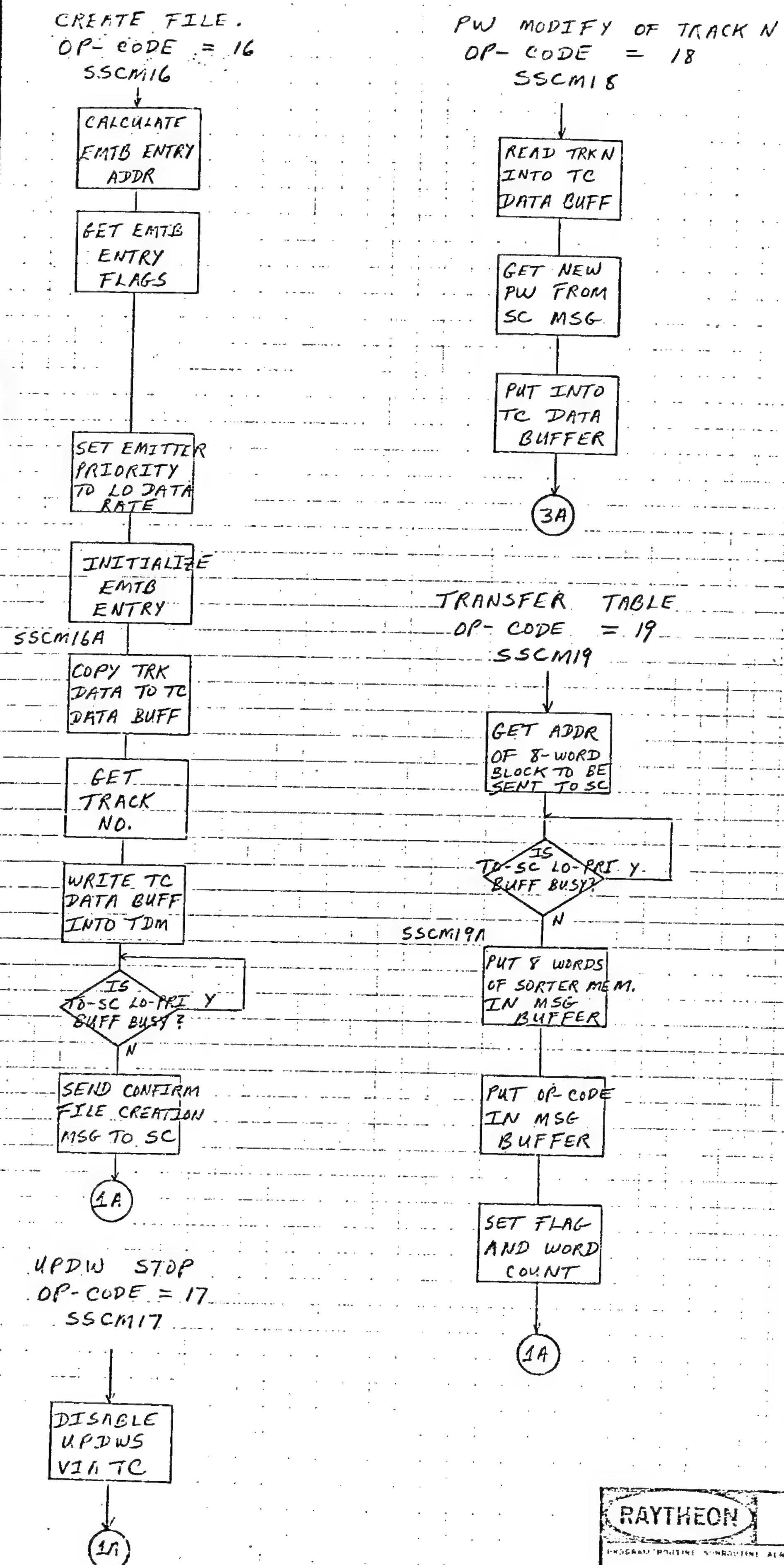
RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV
SC MESSAGE HANDLING

CODE IDENT NO	PREPARED BY	DATE
49956	T. CHERNESKY	12 APR 76

NUMBER 37 | SHEET 6 OF 8

10-2319 (8-73)



REMARKS

RAYTHEON — RAYTHEON COMPANY, LEXINGTON, MASS. 02173

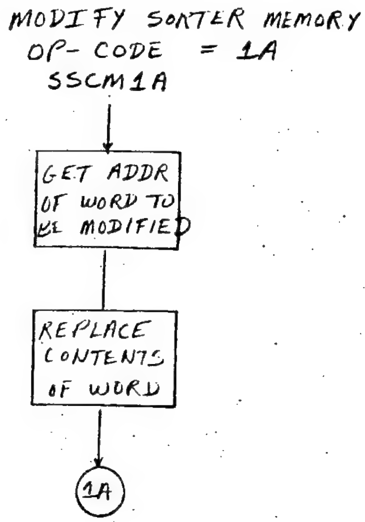
PROGRAM/ROUTINE/SUBROUTINE ACRONYM: IEWS SORTER SUPV

SC MESSAGE HANDLING

49956 | PREPARED BY: T. CHERNESKY | DATE: 3 JUN 76

NUMBER: 33 | SHEET 7 OF 7

FLOW CHART	REMARKS

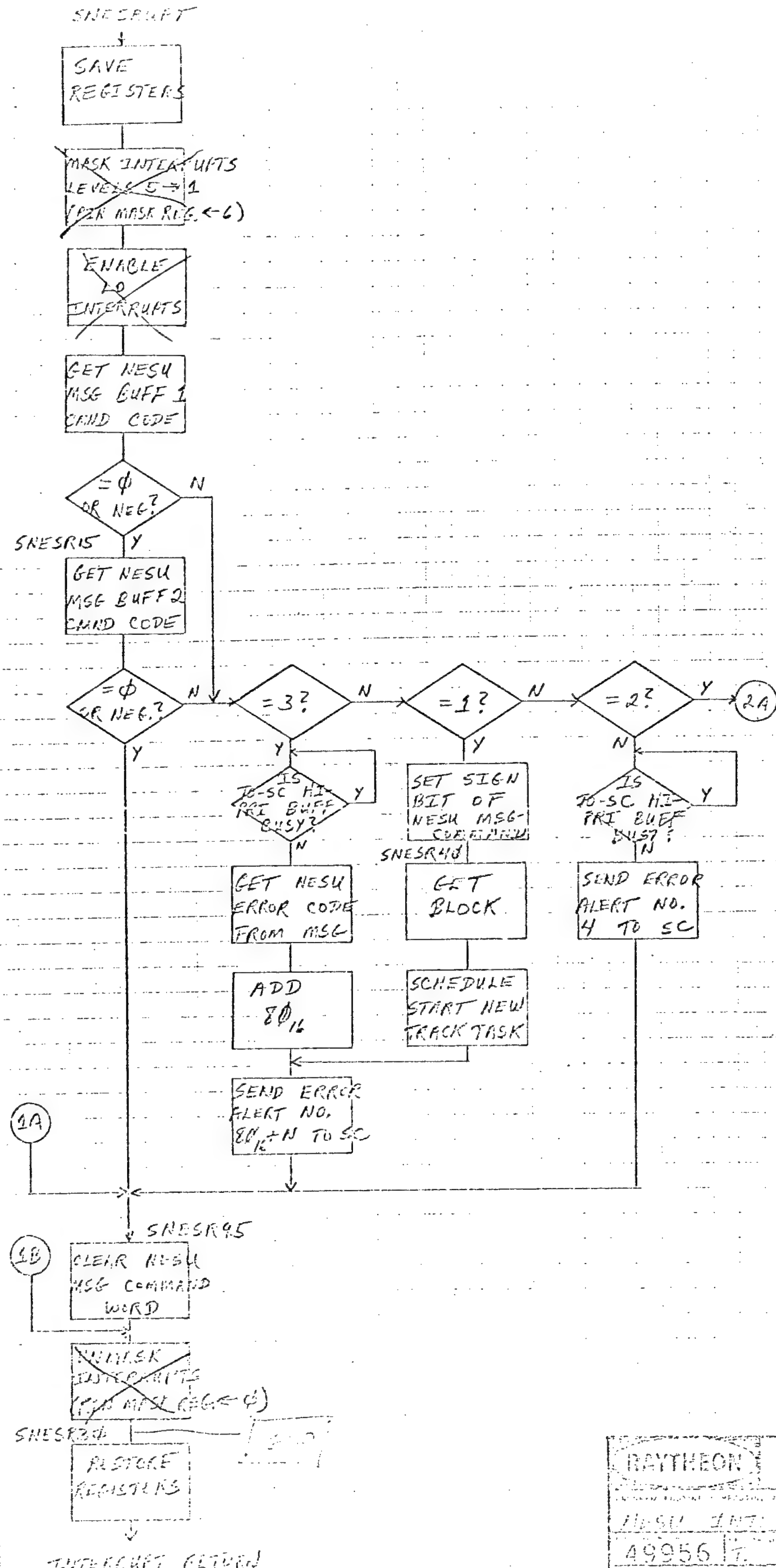


RAYTHEON RAYTHEON COMPANY
LEXINGTON, MASS 02173

PROGRAM/ROUTINE/SUBROUTINE ACRONYM: IEWS SORTER SUPV SC MESSAGE HANDLING

CODE IDENT NO	PREPARED BY	DATE
49956	T. CHERNESKY	12 APR 76

NUMBER 39 SHEET 8 OF 8

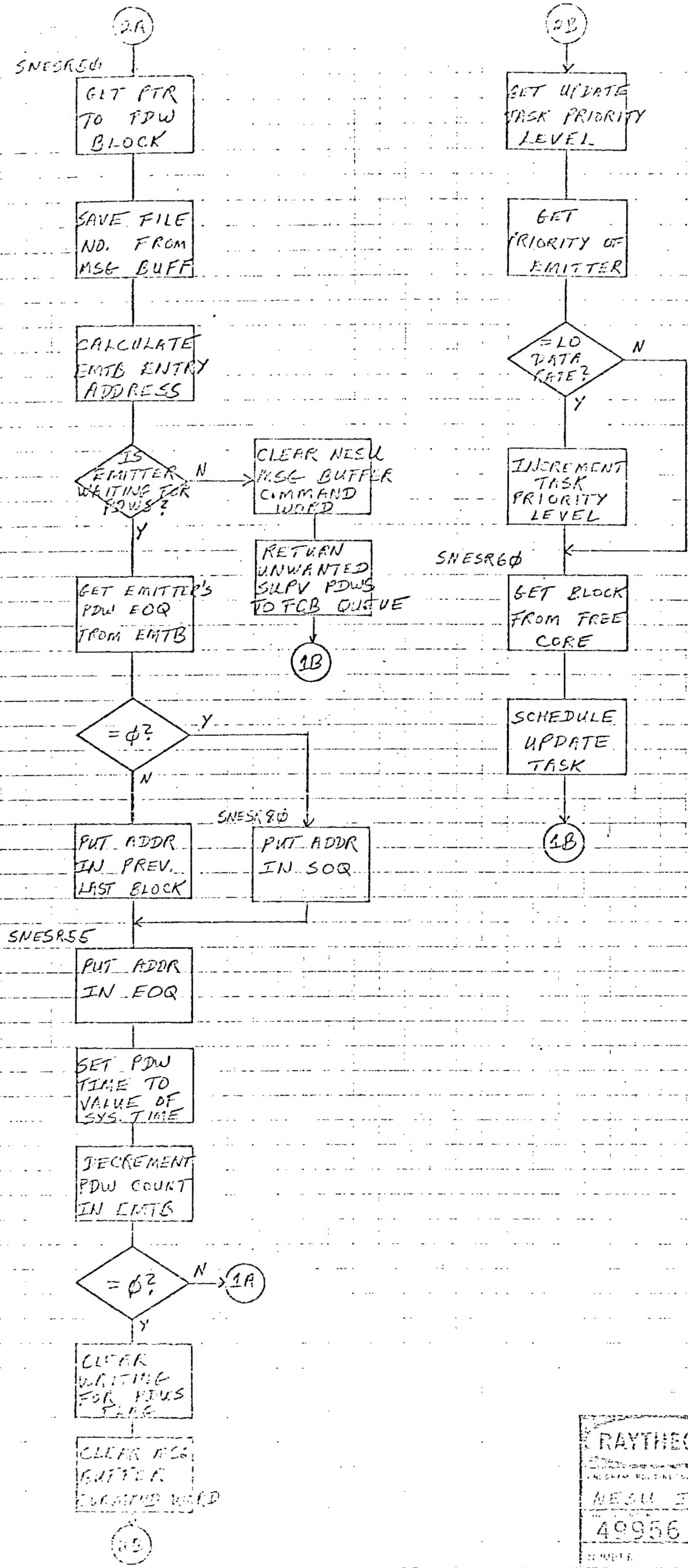


NEGATIVE COMMAND CODE MEANS NEW EMITTER ALERT MSG HAS BEEN PUT IN THIS BUFFER BY NESU AND IS BEING PROCESSED. (I.E., START NEW TRACK HAS BEEN SCHEDULED.)

ERROR ALERT NO. 4 MEANS "INVALID MSG REC'D FROM NESU"

RAYTHEON COMPANY
LEXINGTON MASS 02173

49956



RAYTHEON

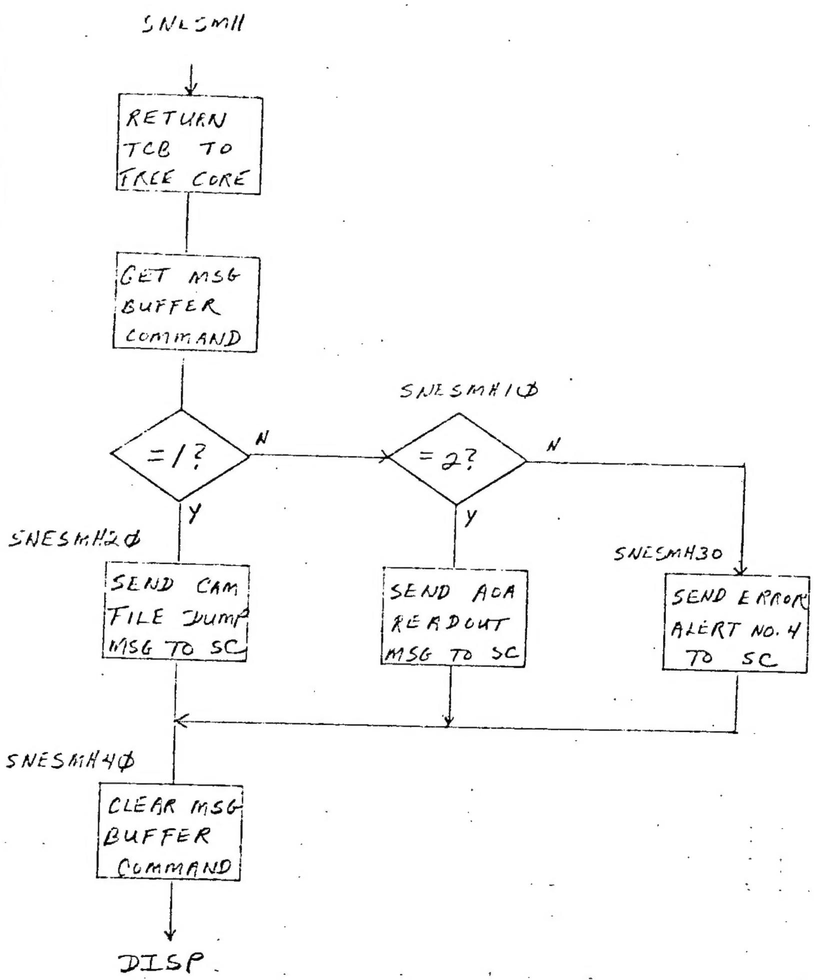
RAYTHEON COMPANY

LEXINGTON, MASS 02173

NESU INTERRUPT HANDLER

49956

56



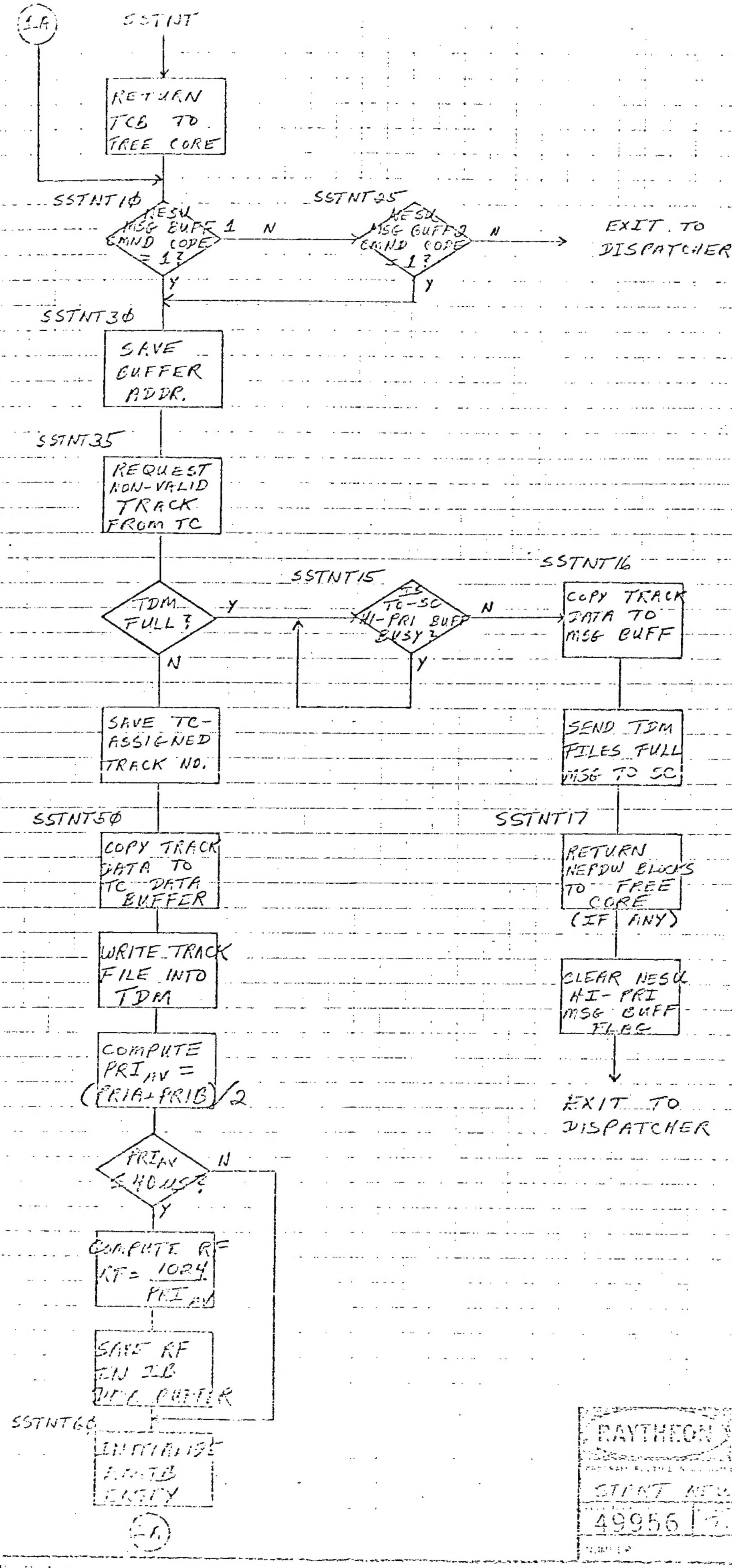
RAYTHEON | RAYTHEON COMPANY LEXINGTON, MASS 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM IEWS SORTER SUPV

NESU MESSAGE HANDLER

CODE IDENT NO 49956 | PREPARED BY T. CHERNESKY | DATE 12 APR 76

NUMBER 28 | SHEET 1 OF 1



$PRI_{AV} \leq 40\mu s$ MEANS THROTTLING OF NEW EMITTER IS REQUIRED.

'RF' MEANS 'REDUCTION FACTOR'

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

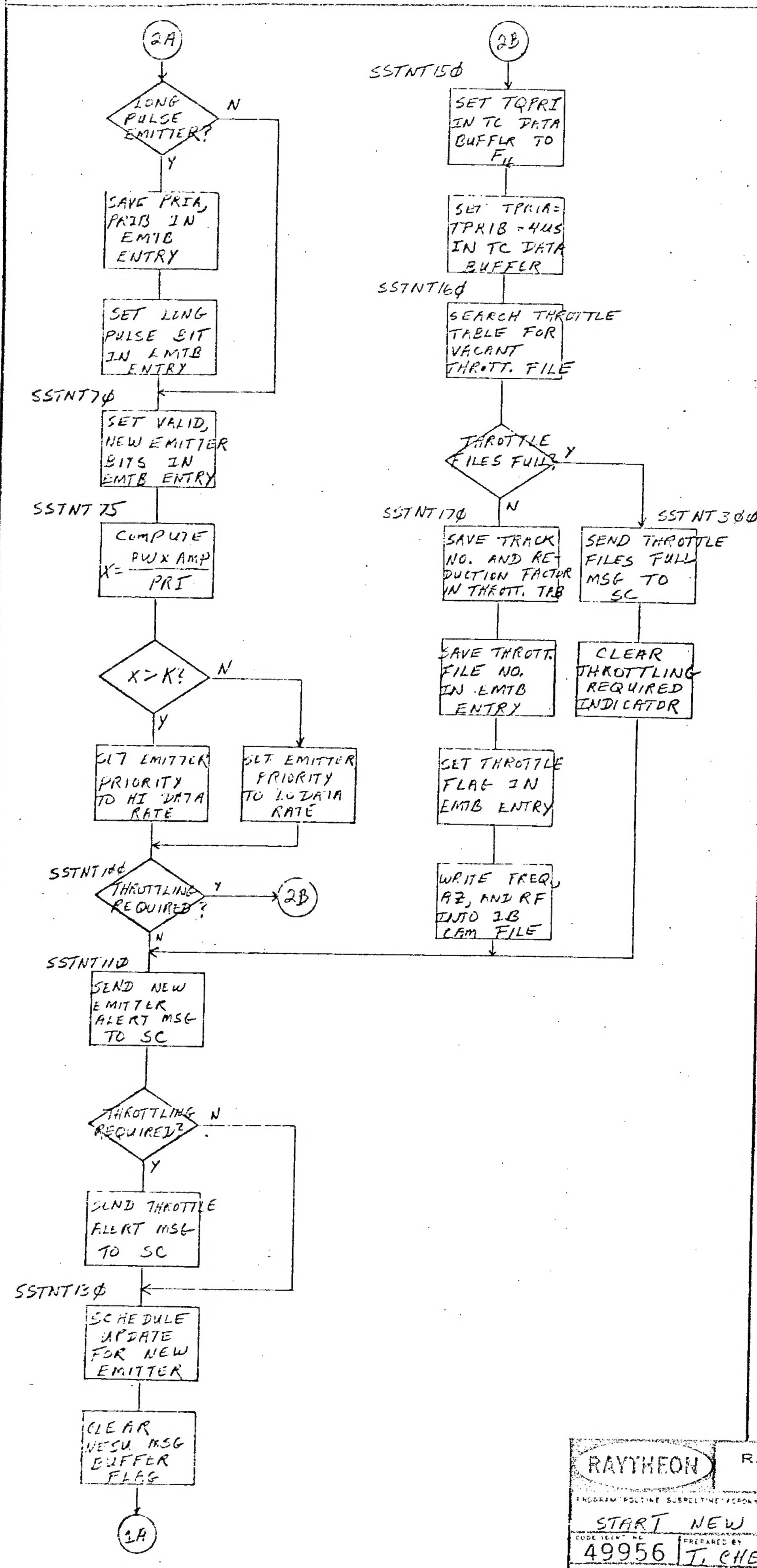
START NEW TRACK TASK

49956

1 OF 2

FLOW CHART

REMARKS



RAYTHEON

RAYTHEON COMPANY
LEXINGTON, MASS. 02173

PROGRAM/ROUTINE/SUBROUTINE/ACRONYM: IEWS SORTER SUPV

START NEW TRACK TASK

CODE IDENT NO: 49956 | PREPARED BY: T. CHERNESKY | DATE: 3 MAY 76

NUMBER: 3Ø | SHEET 2 OF 2